1966-05

Analysis of strain gage data from submarine MSW piping.

Koch, Joseph William Jr.

Monterey, California. U.S. Naval Postgraduate School

ANALYSIS OF STRAIN GAGE DATA FROM SUBMARINE MSW PIPING

JOSEPH WILLIAM KOCH, JR.

5'3

[redacted]

ANALYSIS OF STRAIN GAGE DATA FROM SUBMARINE MSW PIPING

by

Joseph William Koch, Jr.
Lieutenant, United States Navy
B.S., United States Naval Academy, 1957

Submitted in partial fulfillment
for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL
May 1966

ABSTRACT

This thesis presents a procedural sequence for processing and assessing the integrity of experimental strain gage data taken on piping in which gage rosettes are arranged in circumferential patterns at several sections of a piping system. The method of least squares is used to obtain "best" values for strain per unit loading for each gage element. This data is used to find principal stresses and values of normal and shearing stress on a normal cross section at each rosette location. The data is then smoothed to obtain normal and shearing stress as functions of angular position for each ring of rosettes. A computer program has been devised to perform these operations and provide a graphical output to assess degree of fit with experimental data. Finally, manual computation is employed to study the equilibrium of various segments of the piping in evaluating the overall integrity of the data. This procedure and program have been used in evaluating tests performed on the main seawater piping system of the USS Benjamin Franklin (SSBN 640). Some results of the dockside hydrostatic tests are presented herein for purposes of illustration.

TABLE OF CONTENTS

1. Introduction.

ORIGIN OF IDEA FOR THESIS

The idea for this thesis was conceived while the writer worked at the Electric Boat Division of General Dynamics Corporation during an industrial tour from the U. S. Naval Postgraduate School.

While working on the analysis of strain gage data taken during dockside hydrostatic tests and sea trials of the USS Casimir Pulaski (SSBN 633), it became evident that a method to fully analyze the main seawater piping system from the strain gage data did not exist. The only tool available at the time was a computer program that had been developed by Electric Boat which determined the two principal stresses and principal stress axes for each individual set of rosette readings.

The project of instrumenting the main seawater piping system on the USS Benjamin Franklin (SSBN 640) and taking strain gage data during dockside hydrostatic tests and sea trials had been assigned to Electric Boat by the U. S. Navy Bureau of Ships, but no assignment had been made to analyze the data taken. The writer obtained the necessary permission from Electric Boat and the Office of the U. S. Navy Supervisor of Shipbuilding, Groton, Connecticut, to use the data in connection with a thesis at the U. S. Naval Postgraduate School and established liaison with personnel at Electric Boat connected with conducting the SSBN 640 tests.

Over the last nine months Mr. Carl P. Wilson, Supervisor of the piping section of the Marine Engineering Design Department at Electric Boat and his main assistant, Mr. William R. Hintz, have been most helpful and prompt in forwarding the test data and other related material requested. Without their assistance and cooperation this thesis could not have been possible. The writer would like to express his sincere appreciation to Dr. John E. Brock, Professor of Mechanical Engineering, U. S. Naval

Postgraduate School, for his continued patience, encouragement, and most capable guidance while acting as faculty advisor.

The intention of this thesis is not only to analyze the data from the SSBN 640, but also to develop a procedural sequence for processing and assessing the integrity of strain gage data that could be utilized in the design and testing of future submarine seawater piping systems.

HOW DATA WAS OBTAINED

Two complete sets of data were used in developing this thesis. The first set of data was taken on 7 August, 1965 during the dockside hydrostatic test of the main seawater piping system of SSBN 640 in accordance with Electric Boat Division's Propulsion Plant Test Form Number 721-5 Revision A. The second set of data was taken during builder's sea trials on 19 and 20 September, 1965 in accordance with Electric Boat Division's Sea Trial Form Number 21.04 Revision A. However, for security reasons, no further reference is made in this theses to sea trial data.

All strain gages utilized were BLH Type FABR-25-12S9, constantan grid, phenolic glass base, three element 45° superimposed grid rosettes. More detailed information concerning these rosettes may be obtained from BLH Specifications Catalog Bulletin 101 effective September 1964.

Gages were equally spaced around a cross section of piping at 30° from each other. A cross section with twelve rosettes will be referred to as a ring or rosette ring throughout the remainder of this thesis. The rosettes in each ring were oriented as shown in Figure 1.2.1. There were eight such rings.

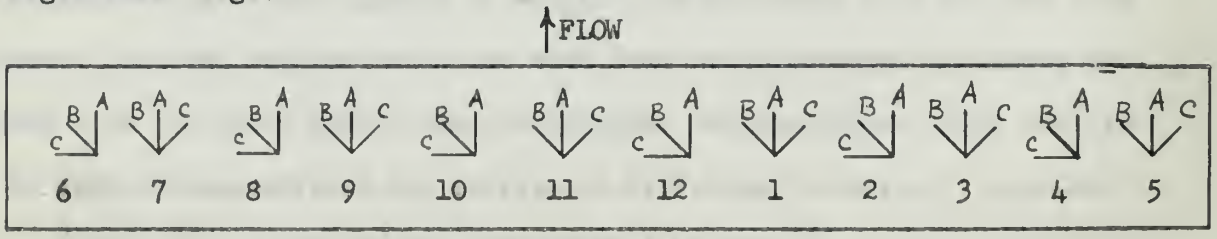


Figure 1.2.1

The number twelve gage is located facing directly up for all rings except ring number 2, in which it faces port. The approximate locations of the rosette rings are shown in figures 1.2.2, 1.2.3 and 1.2.4 which are sketches of the piping system. The exact location of each ring and the piping system dimensions are not given. The dimensions may be found on Bureau of Ships Blueprints SSBN 640 209 2117801 Rev. H and SSBN 640 209 2117802 Rev. F and the exact ring locations may be obtained from either Electric Boat Division's Propulsion Plant Test Form Number 721-5 Rev. A or Sea Trial Data Form Number 21.04 Rev. A.

Strain gage readings were taken using SR-4 Strain Indicators Type N manufactured by Baldwin-Lima-Hamilton Corporation.

OBJECTIVES OF THESIS

The basic purpose of this thesis is to develop methods of assessing the integrity of the data and, incidentally, to reduce the data to a form more convenient for subsequent analysis. To reach this goal a digital computer program has been developed and employed. This program uses as imputs certain constants concerning the piping material and sizes and the raw strain gage data. It computes the two principal stresses, the angle measured from the direction of flow through the piping of the maximum principal stress, and the shear stress at each cross section and stress normal to each cross section. In addition, smoothed values of normal and shear stress every..01 radian around the circumference of the rosette rings are calculated and plots are drawn showing both smoothed and actual data points. Finally the program computes the three forces and moments at each rosette ring.

The outputs of the computer program are analyzed to determine the integrity of the raw data, if the maximum principal stress exceeds yield strength, the agreement between smoothed and actual data values of

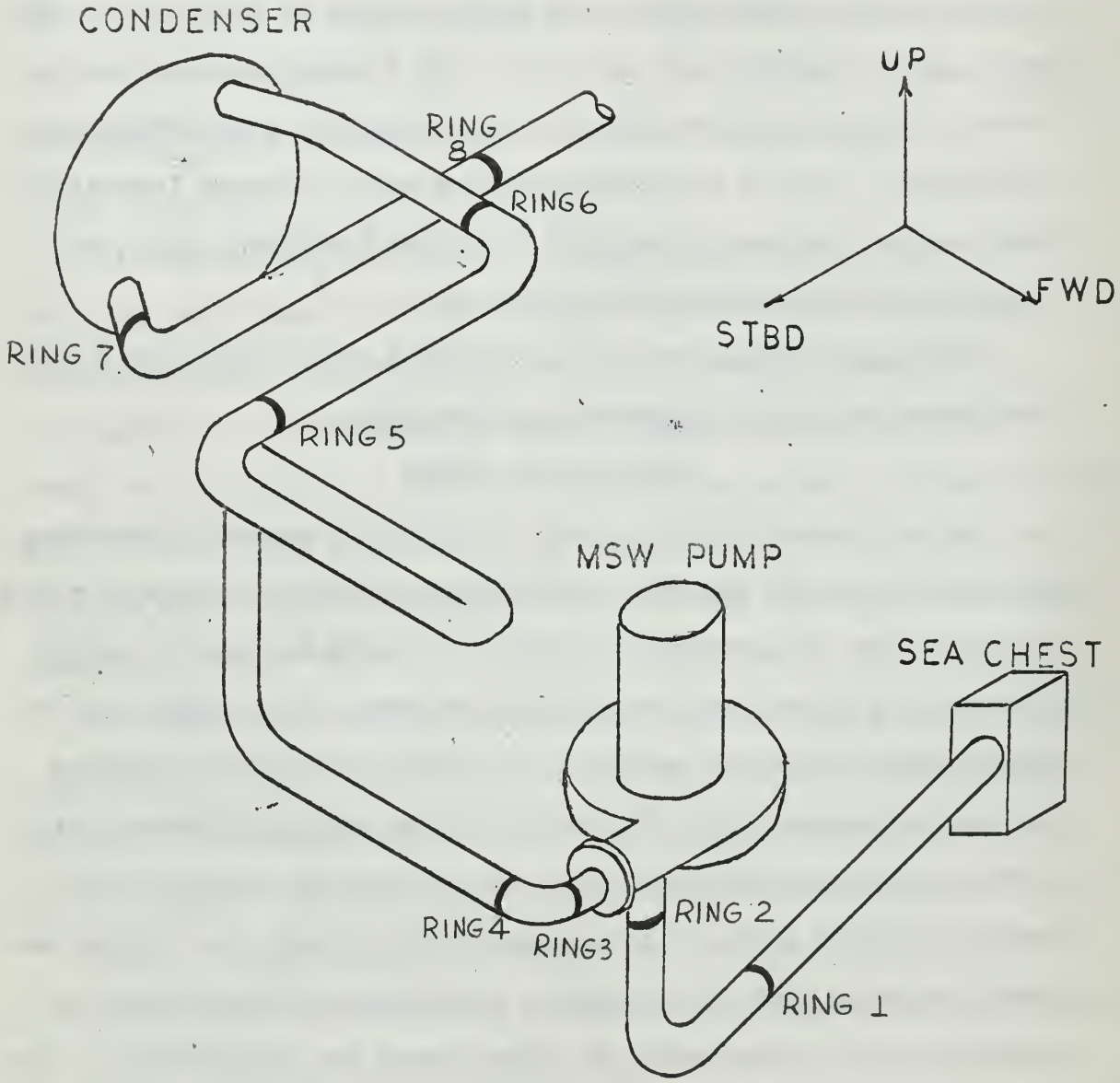
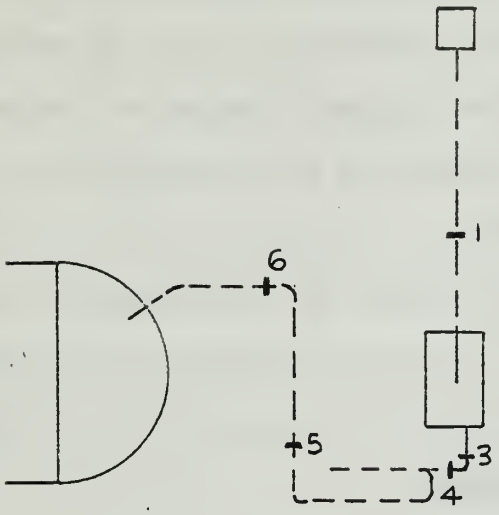
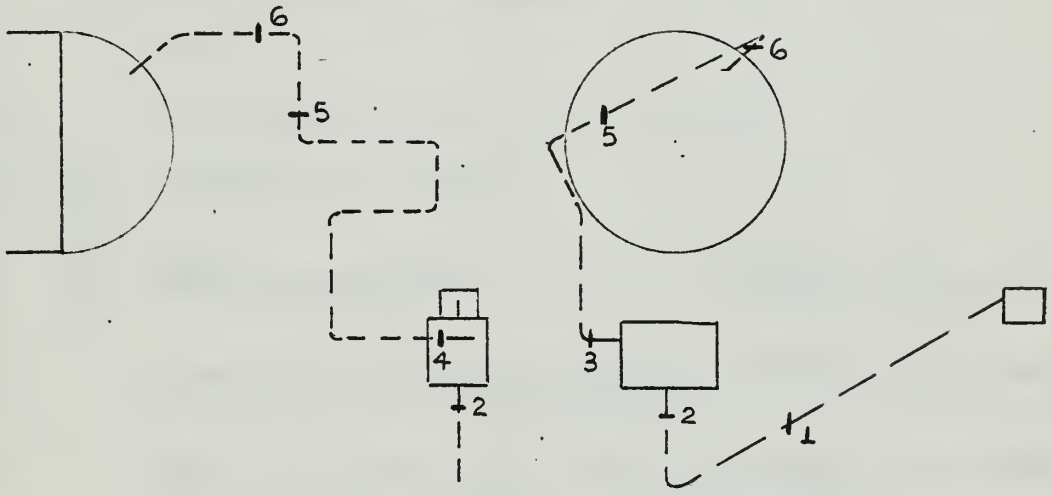

CONDENSER
UP
RING 8
RING 6
FWD
STBD
RING 7
RING 5
MSW PUMP
SEA CHEST
RING 4
RING 3
RING 2
RING 1

FIGURE 1.2.2

MAIN CONDENSER INLET PIPING



TOP VIEW LOOKING DOWN

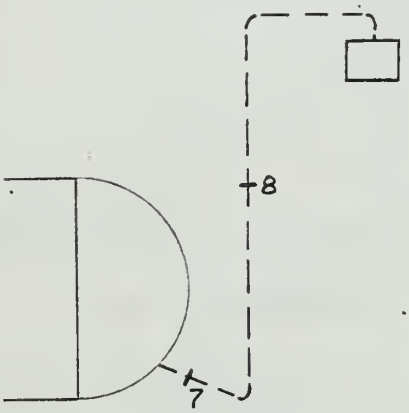


SIDE VIEW LOOKING TO PORT

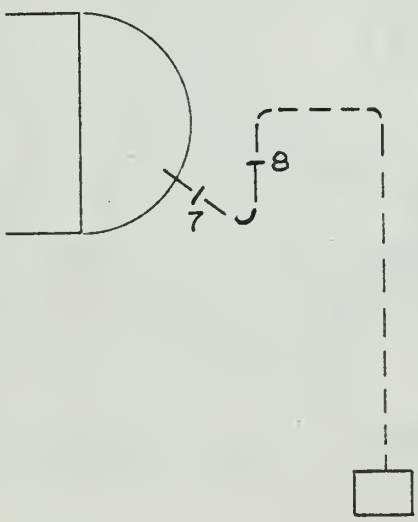
FRONT VIEW LOOKING AFT

FIGURE 1.2.3

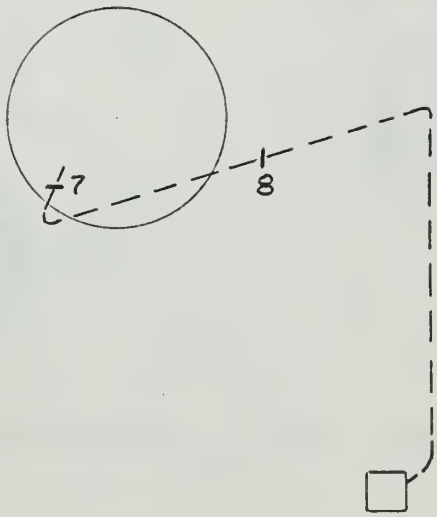
MAIN CONDENSER OUTLET PIPING



TOP VIEW LOOKING DOWN



SIDE VIEW LOOKING TO PORT



FRONT VIEW LOOKING AFT

FIGURE 1.2.4

normal and shear stress and if equilibrium between two rosette rings is established.

For the following sections, the terms experimental data and experimental results are used to represent the actual strain gage data taken and the computed results from this data.

Actual dimensions of the piping system, data, and actual numerical results have been eliminated from this thesis in order to keep the thesis unclassified. All numerical results given are the actual results multiplied by arbitrary constants. However, relative values of the results are the same as the actual results.

NOMENCLATURE

a,b,c,d,e,f,g	Constants for smoothed value of normal stress
E	Young's Modulus
e_i	Error
$\bar{e}_\eta, \bar{e}_\varphi$	Unit vectors in the η and φ directions
$F(\phi_i), F_1(\phi_i)$	Functions of the angle ϕ
F_{Rx}, F_{RY}, F_{Rz}	Resultant equilibrium forces in the x,y and z directions
F_x	Force on a cross section of the main seawater piping system in the direction of the ship's bow.
F_Y	Force on a cross section of the main seawater piping system in the vertical direction from ship's centerline.
F_z	Force on a cross section of the main seawater piping system in the starboard direction from ship's centerline.
$F_{x'}$, $F_{Y'}$, $F_{z'}$	Forces in the x' , y' , and z' directions
$\bar{i},\bar{j},\bar{k}$	Unit vectors in the x, y and z directions.
k,l,m,n,o,p,q	Constants for smoothed value of shearing stress
$x,y,z;x',y',z'$	Coordinate systems described in text

M_{Rx}, M_{Ry}, M_{Rz}	Resultant equilibrium moments in the x, y and z directions.
M_x	Moment about the fore and aft axis of the ship in the direction of the bow.
M_Y	Moment about the vertical axis of the ship in the upward direction.
M_z	Moment about the port and starboard axis of the ship in the direction of starboard
$M_{x'}$, $M_{Y'}$, $M_{z'}$	Moments in the x' , y' , and z' directions.
r	Radius of the piping.
t	Pipe wall thickness.
x	percent of test depth.
y	Strain reading at any depth
[ ]	Matrix
[A]	a 7 x 7 matrix.
[B] , [B'], [C] , [C']	7 x 1 Matrices
$\epsilon_1, \epsilon_2, \epsilon_3$	Three strains of a three element 90° rosette.
Θ_p	Maximum principal stress angle
μ	Poisson's ratio.
Σ	Summation sign
σ_{max}	Maximum principal stress
σ_{min}	Minimum principal stress
σ_N	Stress normal to a cross section of piping in the direction of flow.
τ_{xY}	Shear stress on the outside surface of the pipe positive in the direction of increasing rosette numbers.
τ_{xz}	Shear stress in the xz plane
ϕ	Angle on piping cross section from rosette number 12.
$\mathcal{S}$	Stress tensor (dyadic).

2. Theory[1]

THEORY FOR STRAIGHT LINE FITTING

It is assumed that the strain gage readings will increase linearly with internal pressure which in turn does increase linearly with an increase in depth. This assumption is based on the following facts:

(a) The piping material exhibits a straight line stress-strain curve.

(b) The system is not loaded above the elastic limit of the material as verified by the maximum values of the maximum principal stress at every rosette.

(c) The deflections are small as verified by the overall change in strain gage readings.

(d) It is assumed, for the builder's sea trials, that hull deflection inputs to the piping increase linearly with depth. For dockside hydrostatic tests, there was no such input.

The other main effect acting on the system is the weight of the system. Since the system is always filled with seawater of nearly constant density the weight effect on the piping system will remain nearly constant and, therefore, not have any appreciable effect on the assumption of linearity.

The method of least squares[2] is used to obtain the best straight line connecting all data points of each element with the percentage of test depth taken as the independent variable and the strain gage readings taken as the dependent variable. The results of the method of least squares are given in the form of the equation:

[1]Greater details of the theoretical relations are given in, or may easily be inferred from, corresponding subsection of section 3, which deals with the computer program used to perform the calculations.

[2]Spiegel, M.R., Schaum's Outline Series, Theory and Problems of Statistics (New York; Schaum Publishing Co., 1961) p.220

$$Y = a + bx \qquad 2.1.1$$

where a is the zero percent test depth intercept and b is the slope of the best straight line connecting all data points from the surface to test depth. The slope, b, is also proportional to change in strain per percent change in test depth. The slopes are the values of the individual element strains, ϵ_1, ϵ_2, and ϵ_3, used in the subsequent development of the theory.

THEORY FOR SOLVING PRINCIPAL STRESSES AND ANGLES

The equations defining the principal stresses in terms of the strains measured by a rectangular rosette are[3]:

$$\sigma_{max} = \frac{E}{2}\left\{\frac{\epsilon_1+\epsilon_3}{1-\mu} + \frac{1}{1+\mu}\sqrt{(\epsilon_1-\epsilon_3)^2 + [2\epsilon_2 - (\epsilon_1+\epsilon_3)]^2}\right\} \qquad 2.2.1$$

$$\sigma_{min} = \frac{E}{2}\left\{\frac{\epsilon_1+\epsilon_3}{1-\mu} - \frac{1}{1+\mu}\sqrt{(\epsilon_1-\epsilon_3)^2 + [2\epsilon_2 - (\epsilon_1+\epsilon_3)]^2}\right\} \qquad 2.2.2$$

where the elements of the rosette are numbered as shown in figure 2.2.1.

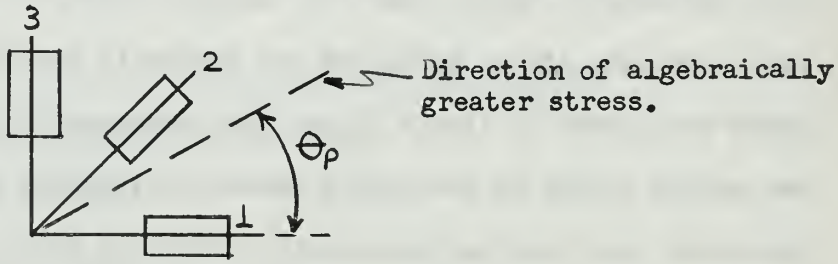


Figure 2.2.1

The direction of the maximum principal stress from element number one of the rosette in terms of the measured strains is:

$$\theta_p = \frac{1}{2}\arctan\frac{2\epsilon_2 - (\epsilon_1+\epsilon_3)}{\epsilon_1-\epsilon_3} \qquad 2.2.3$$

[3]The derivation of equations 2.2.1, 2.2.2, and 2.2.3 is given in Perry and Lissner, The Strain Gage Primer, pages 136-138

THEORY FOR EXPERIMENTAL VALUES OF σ_N & τ_{xy}

Looking down on the surface of the piping, the stress picture looks like figure 2.3.1.

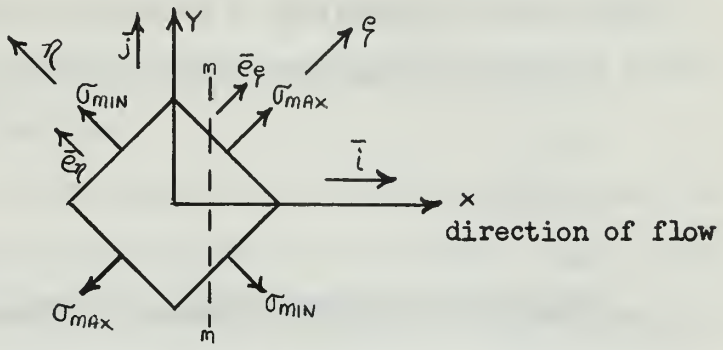


Figure 2.3.1

It is known that σ_3 which is perpendicular to the plane of σ_1 and σ_2 is equal to zero.

To determine the stresses on a plane perpendicular to the x-y plane whose trace is mm, tensor theory is used. Temporarily use coordinates ξ, η, and z to determine σ_N, τ_{xy} and τ_{xz}. (The direction z is perpendicular to the plane shown in figure 2.3.1 with unit vector $\bar{k}$.)

From tensor theory, the stress tensor (dyadic) may be written:

$$\mathcal{S} = \sigma_{MAX}\,\bar{e}_\xi\bar{e}_\xi + \sigma_{MIN}\,\bar{e}_\eta\,\bar{e}_\eta \qquad 2.3.1$$

$$\sigma_N = \bar{i}\cdot\mathcal{S}\cdot\bar{i} \qquad 2.3.2$$

$$\tau_{xy} = \bar{i}\cdot\mathcal{S}\cdot\bar{j} \qquad 2.3.3$$

$$\tau_{xz} = \bar{i}\cdot\mathcal{S}\cdot\bar{k} \qquad 2.3.4$$

Since the term $\bar{i}\cdot\mathcal{S}$ appears in equations 2.3.2, 2.3.3, and 2.3.4, solve for it first.

$$\bar{i}\cdot\mathcal{S} = \sigma_{MAX}(\bar{i}\cdot\bar{e}_\xi)\bar{e}_\xi + \sigma_{MIN}(\bar{i}\cdot\bar{e}_\eta)\,\bar{e}_\eta$$

$$= \sigma_{MAX}\cos\theta_p\,\bar{e}_\xi - \sigma_{MIN}\sin\theta_p\,\bar{e}_\eta \qquad 2.3.5$$

Substituting equation 2.3.5 back into equations 2.3.2, 2.3.3, and 2.3.4 results in the following equations:

$$\sigma_N = \sigma_{MAX}\cos\theta_p\,(\bar{e}_\xi\cdot\bar{i}) - \sigma_{MIN}\sin\theta_p\,(\bar{e}_\eta\cdot\bar{i})$$

$$= \sigma_{MAX}\cos^2\theta_p + \sigma_{MIN}\sin^2\theta_p \qquad 2.3.6$$

$$\tau_{xy} = \sigma_{MAX} \cos\theta_p (\bar{e}_\xi \cdot \bar{j}) - \sigma_{MIN} \sin\theta_p (\bar{e}_\eta \cdot \bar{j})$$

$$= \sigma_{MAX} \cos\theta_p \sin\theta_p - \sigma_{MIN} \sin\theta_p \cos\theta_p$$

$$= \tfrac{1}{2}(\sigma_{MAX} - \sigma_{MIN}) \sin 2\theta_p \qquad 2.3.7$$

$$\tau_{xz} = \sigma_{MAX} \cos\theta_p (\bar{e}_\xi \cdot \bar{k}) - \sigma_{MIN} \sin\theta_p (\bar{e}_\eta \cdot k)$$

$$= (\sigma_{MAX} \cos\theta_p)\, 0 - (\sigma_{MIN} \sin\theta_p)\, 0$$

$$= 0 \qquad 2.3.8$$

THEORY FOR SMOOTHED VALUES OF NORMAL AND SHEARING STRESS

In developing the theory for the smoothed values of σ_N and τ_{xy}, it has been assumed that the radius of the piping has remained constant throughout the entire piping system. This assumption is based on the diameter readings taken by GD/EB of rosette rings two through eight. These readings indicated that the diameter of the piping around a rosette ring remained constant enough to use a constant value of radius for calculations in this thesis.

Since a previously developed theory was not available, it seemed reasonable to assume that σ_N and τ_{xy} vary only with ϕ in the following Fourier Series:

$$\sigma_N = a + b\sin\phi + c\cos\phi + d\sin 2\phi + e\cos 2\phi + f\sin 3\phi$$

$$+ g\cos 3\phi + \cdots = F(\phi) \qquad 2.4.1$$

$$\tau_{xy} = k + l\sin\phi + m\cos\phi + n\sin 2\phi + o\cos 2\phi + p\sin 3\phi$$

$$+ q\cos 3\phi + \cdots = F_1(\phi) \qquad 2.4.2$$

One of the decisions that had to be made concerned the number of terms of the Fourier Series to use. Since curves showing the exact shape of the stress distribution around the circumference did not exist, there was no criterion available to which curves developed could be compared. To assist in deciding how many terms to use, the solution was investigated using five, seven, nine, and eleven terms. A plot of the

experimental values and the smoothed values of σ_N for five, seven, nine, and eleven terms for rosette ring number three is shown in figure 2.4.1. A decision was made to use seven terms for the following reasons:

(a) A certain amount of redundancy was necessary to permit good least squares evaluation.

(b)If higher coefficients were really of significance, more than twelve rosettes would be required at each rosette ring to obtain more than seven coefficients whose accuracy could not be doubted.

(c) For equilibrium, only the first three coefficients are used. Therefore, it is necessary to have these evaluated as well as possible.

The truncated forms of equations 2.4.1 and 2.4.2 shown below will be used in the subsequent development of theory.

$$\sigma_N = a + b\sin\phi + c\cos\phi + d\sin 2\phi + e\cos 2\phi + f\sin 3\phi + g\cos 3\phi = F(\phi) \qquad 2.4.1a$$

$$\tau_{xy} = k + l\sin\phi + m\cos\phi + n\sin 2\phi + o\cos 2\phi + p\sin 3\phi + q\cos 3\phi = F_1(\phi) \qquad 2.4.2a$$

The smoothed values of σ_N will be discussed before proceeding with the smoothed values of τ_{xy}.

There are N=12 stations around the circumference of the pipe for which there are experimental values of σ_N, i.e., there exists a σ_i corresponding to $F(\phi_i)$, i=1,...,N. σ_i is considered to be the experimental value if $F(\phi_i)$ is considered to be the smoothed value. It is assumed that:

$$\text{error} = e_i = \sigma_i - F(\phi_i) \qquad 2.4.3$$

It is necessary to minimize the sum of the square of the errors in order to have smoothed results and experimental results within reasonable agreement. Therefore,

$$\varepsilon = \Sigma (e_i)^2$$

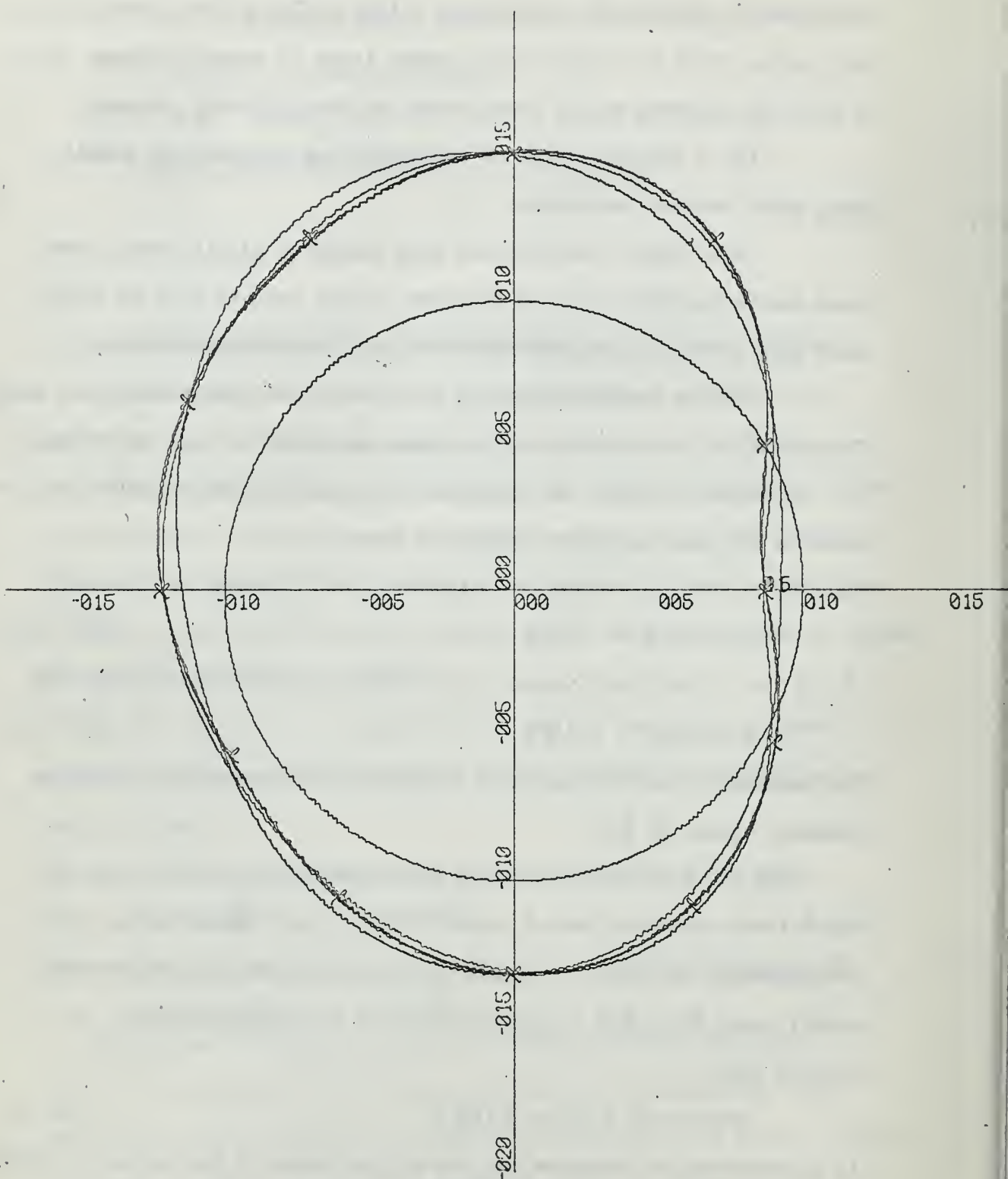


FIGURE 2.4.1

is to be minimized. This means:

$$\left.\begin{aligned}
&\frac{\partial E}{\partial a} = 0\\
&\frac{\partial E}{\partial b} = 0\\
&\frac{\partial E}{\partial c} = 0\\
&\frac{\partial E}{\partial d} = 0\\
&\frac{\partial E}{\partial e} = 0\\
&\frac{\partial E}{\partial f} = 0\\
&\frac{\partial E}{\partial g} = 0
\end{aligned}\right\} \quad 2.4.4$$

Equations 2.4.4 are solved in the following manner:

$$e_i = \sigma_i - F(\phi_i)$$

$$= \sigma_i - [a + b\sin\phi + c\cos\phi + d\sin 2\phi + e\cos 2\phi + f\sin 3\phi + g\cos 3\phi] \quad 2.4.5$$

$$E = \Sigma(e_i)^2 = \sum_{i=1}^{N} \{\sigma_i^2 - 2\sigma_i F(\phi_i) + [F(\phi_i)]^2\}$$

$$\frac{\partial E}{\partial K} = -2\sum_{i=1}^{N} \sigma_i \frac{\partial F(\phi_i)}{\partial K} + 2F(\phi_i)\frac{\partial F(\phi_i)}{\partial K} = 0 \quad 2.4.6$$

where K = a, b, c, d, e, f, g

By substituting the various values of K in equation 2.4.6 the following equations are obtained:

$$\frac{\partial E}{\partial a} = -2\Sigma\sigma_i + 2\Sigma F(\phi_i) = 0 \quad 2.4.7$$

$$\frac{\partial E}{\partial b} = -2\Sigma\sigma_i \sin\phi_i + 2\Sigma F(\phi_i)\sin\phi_i = 0 \quad 2.4.8$$

$$\frac{\partial E}{\partial c} = -2\Sigma\sigma_i \cos\phi_i + 2\Sigma F(\phi_i)\cos\phi_i = 0 \quad 2.4.9$$

$$\frac{\partial E}{\partial d} = -2\Sigma\sigma_i \sin 2\phi_i + 2\Sigma F(\phi_i)\sin 2\phi_i = 0 \quad 2.4.10$$

$$\frac{\partial E}{\partial e} = -2\Sigma\sigma_i \cos 2\phi_i + 2\Sigma F(\phi_i)\cos 2\phi_i = 0 \quad 2.4.11$$

$$\frac{\partial E}{\partial f} = -2\Sigma\sigma_i \sin 3\phi_i + 2\Sigma F(\phi_i)\sin 3\phi_i = 0 \qquad 2.4.12$$

$$\frac{\partial E}{\partial g} = -2\Sigma\sigma_i \cos 3\phi_i + 2\Sigma F(\phi_i)\cos 3\phi_i = 0 \qquad 2.4.13$$

Shifting the first terms to the right hand side and substituting equation 2.4.1a for $F(\phi_i)$ in equations 2.4.7 through 2.4.13, the results displayed in matrix form are:

$$[A]\,[B] = [C] \qquad 2.4.14$$

where:

$$[A] = \begin{bmatrix} N & \Sigma s\phi_i & \Sigma c\phi_i & \Sigma s2\phi_i & \Sigma c2\phi_i & \Sigma s3\phi_i & \Sigma c3\phi_i \\ \Sigma s\phi_i & \Sigma s^2\phi_i & \Sigma s\phi_i c\phi_i & \Sigma s\phi_i s2\phi_i & \Sigma s\phi_i c2\phi_i & \Sigma s\phi_i s3\phi_i & \Sigma s\phi_i c3\phi_i \\ \Sigma c\phi_i & \Sigma c\phi_i s\phi_i & \Sigma c^2\phi_i & \Sigma c\phi_i s2\phi_i & \Sigma c\phi_i c2\phi_i & \Sigma c\phi_i s3\phi_i & \Sigma c\phi_i c3\phi_i \\ \Sigma s2\phi_i & \Sigma s2\phi_i s\phi_i & \Sigma s2\phi_i c\phi_i & \Sigma s^2 2\phi_i & \Sigma s2\phi_i c2\phi_i & \Sigma s2\phi_i s3\phi_i & \Sigma s2\phi_i c3\phi_i \\ \Sigma c2\phi_i & \Sigma c2\phi_i s\phi_i & \Sigma c2\phi_i c\phi_i & \Sigma c2\phi_i s2\phi_i & \Sigma c^2 2\phi_i & \Sigma c2\phi_i s3\phi_i & \Sigma c2\phi_i c3\phi_i \\ \Sigma s3\phi_i & \Sigma s3\phi_i s\phi_i & \Sigma s3\phi_i c\phi_i & \Sigma s3\phi_i s2\phi_i & \Sigma s3\phi_i c2\phi_i & \Sigma s^2 3\phi_i & \Sigma s3\phi_i c3\phi_i \\ \Sigma c3\phi_i & \Sigma c3\phi_i s\phi_i & \Sigma c3\phi_i c\phi_i & \Sigma c3\phi_i s2\phi_i & \Sigma c3\phi_i c2\phi_i & \Sigma c3\phi_i s3\phi_i & \Sigma c^2 3\phi_i \end{bmatrix} \qquad 2.4.15$$

(In this matrix only, s is used for sin, c is used for cos.)

$$[B] = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \end{bmatrix} \qquad 2.4.16$$

$$[C] = \begin{bmatrix} \Sigma\sigma_i \\ \Sigma\sigma_i \sin\phi_i \\ \Sigma\sigma_i \cos\phi_i \\ \Sigma\sigma_i \sin 2\phi_i \\ \Sigma\sigma_i \cos 2\phi_i \\ \Sigma\sigma_i \sin 3\phi_i \\ \Sigma\sigma_i \cos 3\phi_i \end{bmatrix} \qquad 2.4.17$$

When the number of stations around the circumference is even and the stations are equally spaced, it can readily be shown that all non-diagonal elements of the A matrix in equation 2.4.15 become zero. In this case the seven equations with seven common unknowns become seven equations, each with only one unknown, as follows:

$$\left.\begin{aligned} aN &= \Sigma\sigma_i \\ b\Sigma \sin^2\phi_i &= \Sigma\sigma_i \sin^2\phi_i \\ c\Sigma \cos^2\phi_i &= \Sigma\sigma_i \cos^2\phi_i \\ d\Sigma \sin^2 2\phi_i &= \Sigma\sigma_i \sin^2 2\phi_i \\ e\Sigma \cos^2 2\phi_i &= \Sigma\sigma_i \cos^2 2\phi_i \\ f\Sigma \sin^2 3\phi_i &= \Sigma\sigma_i \sin^2 3\phi_i \\ g\Sigma \cos^2 3\phi_i &= \Sigma\sigma_i \cos^2 3\phi_i \end{aligned}\right\} \quad 2.4.18$$

By a similar derivation, the following equations are established for the seven coefficients used in determining the smoothed values of τ_{xy}.

$$[A]\ [B'] = [C'] \qquad 2.4.19$$

where [A] is the same as shown in equation 2.4.15,

$$[B'] = \begin{bmatrix} k \\ l \\ m \\ n \\ o \\ p \\ q \end{bmatrix} \qquad 2.4.20$$

$$[C'] = \begin{bmatrix} \Sigma\tau_i \\ \Sigma\tau_i \sin\phi_i \\ \Sigma\tau_i \cos\phi_i \\ \Sigma\tau_i \sin 2\phi_i \\ \Sigma\tau_i \cos 2\phi_i \\ \Sigma\tau_i \sin 3\phi_i \\ \Sigma\tau_i \cos 3\phi_i \end{bmatrix} \qquad 2.4.21$$

Equations 2.4.17 become:

$$\left.\begin{aligned} kN &= \Sigma\tau_i \\ l\Sigma \sin^2\phi &= \Sigma\tau_i \sin^2\phi_i \\ m\Sigma \cos^2\phi &= \Sigma\tau_i \cos^2\phi_i \\ n\Sigma \sin^2 2\phi &= \Sigma\tau_i \sin^2 2\phi_i \\ o\Sigma \cos^2 2\phi &= \Sigma\tau_i \cos^2 2\phi_i \\ p\Sigma \sin^2 3\phi &= \Sigma\tau_i \sin^2 3\phi_i \\ q\Sigma \cos^2 3\phi &= \Sigma\tau_i \cos^2 3\phi_i \end{aligned}\right\} \quad 2.4.22$$

Finding the seven unknowns for σ_N and the seven unknowns for τ_{xy} by use of equations 2.4.18 and 2.4.22 will be referred to as the short method in the remainder of this thesis. Finding the unknowns for σ_N and τ_{xy} by use of equations 2.4.14 and 2.4.19 will be referred to as the long method in the remainder of this thesis.

Once the two sets of seven unknowns have been found by either the short or long method, they are substituted into equations 2.4.1a and 2.4.2a to determine the smoothed values of σ_N and τ_{xy}.

THEORY FOR EQUILIBRIUM CALCULATIONS

Equilibrium of the system was established in several steps. The first step was to calculate the sum of the forces and moments at each rosette ring cross section. The x′ direction was taken as the direction of flow in the pipe, the y′ direction as the direction from the center of the pipe toward rosette number 12, and the z′ direction as the direction from the center of the pipe toward rosette number 3, as shown in figure 2.5.1.

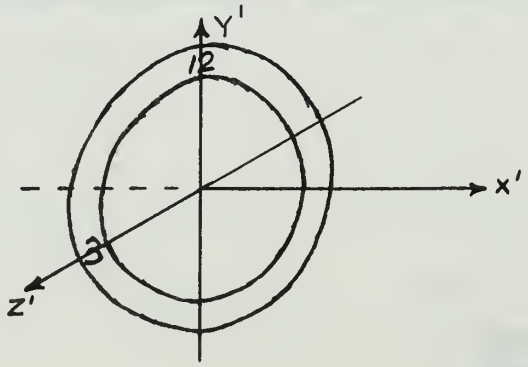


Figure 2.5.1

The equations to obtain the sum of the forces and moments are:

$$F_{x'} = t\int_0^{2\pi} r\,\sigma_N\,d\phi = rt\int_0^{2\pi} (a + b\sin\phi + c\cos\phi + d\sin 2\phi + \cdots)\,d\phi$$

$$= 2\pi a r t \qquad 2.5.1$$

$$F_{y'} = -t\int_0^{2\pi} r\sin\phi\,\tau_{xy}\,d\phi = -rt\int_0^{2\pi} \sin\phi\,(k + l\sin\phi + m\cos\phi + \cdots)\,d\phi$$

$$= -\pi l r t \qquad 2.5.2$$

$$F_z' = t\int_0^{2\pi} r\cos\phi\,\tau_{xy}\,d\phi = rt\int_0^{2\pi}\cos\phi(k+l\sin\phi+m\cos\phi+\cdots)d\phi$$

$$= \pi m r t \qquad 2.5.3$$

$$M_x' = t\int_0^{2\pi} r^2\tau_{xy}\,d\phi = r^2t\int_0^{2\pi}(k+l\sin\phi+m\cos\phi+\cdots)d\phi$$

$$= 2\pi k r^2 t \qquad 2.5.4$$

$$M_y' = t\int_0^{2\pi} r^2\sin\phi\,\sigma_N\,d\phi = r^2t\int_0^{2\pi}\sin\phi(a+b\sin\phi+c\cos\phi+\cdots)d\phi$$

$$= \pi b r^2 t \qquad 2.5.5$$

$$M_z' = -t\int_0^{2\pi} r^2\cos\phi\,\sigma_N\,d\phi = r^2t\int_0^{2\pi}\cos\phi(a+b\sin\phi+\cdots)d\phi$$

$$= -\pi c r^2 t \qquad 2.5.6$$

The second step in the equilibrium calculations is to reorient each individual cross section coordinate system to a standard coordinate system. The standard coordinate system aligns the x direction with the fore to aft axis of the ship, the y direction with the vertical upward axis of the ship, and the z direction with the port to starboard axis of the ship. The reorientation is obtained using actual dimensions and angles taken from the system blueprints and standard coordinate reorientation methods.

The third step in the equilibrium establishment consists of making the segments between rosette rings into free bodies and summing the forces and moments acting on the segment. In addition to the forces and moments calculated in equations 2.5.1 through 2.5.6, the forces due to the fluid in the pipe segment are added to obtain the equilibrium forces

and moments. The signs of the forces and moments at the end of the segment in which the fluid enters are reversed because the values calculated in equations 2.5.1 through 2.5.6 are the forces of the adjacent segment acting in that segment and not the forces of the incoming segment acting in the adjacent segment as are desired for equilibrium calculations. Theoretically the sums of the forces and moments on each individual segment should be zero.

Only step one utilizes the computer program developed. Since each rosette ring is oriented differently with respect to the standard coordinate system chosen, more time would be required to adjust the program to calculate the different reorientations for equilibrium calculations than to calculate the reorientations by hand. The same applies to step three. Therefore, steps two and three are performed by hand.

3. Explanation of Computer Program

[illegible]. INTRODUCTORY COMMENTS

This section presents a full explanation of the digital computer program developed for evaluating the strain gage data. A listing of the various symbols used in the program, but not defined in the program explanation, is found in Appendix B. A complete listing of the program appears in Appendix A. The statement numbers referred to in this section are the six digit letters following each line of the program listing.

The program language employed in this program is CDC FORTRAN 60. All computer runs were made on the Control Data Corporation 1604 computer and associated equipment located at the U. S. Naval Postgraduate School at Monterey, California.

The program developed is divided into 12 parts. The following subsections describe these parts.

PART 1, DATA READ IN

Part 1 of the program reads in various quantities that remain constant during the entire program. These quantities most likely would vary if used for another piping system, but would still remain constant for that particular system.

The quantities read in are N, the number of experimental depths at which readings were taken; E, Young's modulus for the pipe material in millions of pounds per square inch; μ, Poisons Ratio for the pipe material used; TH(1) through TH(8), the average wall thicknesses for rings one through eight; RA(1) through RA(8), the pipe radius for ring one through ring eight[4]; and Z, the angle between two strain gage rosettes of each ring in radians. Also included in this part are the commands which

[4]These values of radius and thickness were obtained by averaging the thicknesses and radii at the various measurement points for each ring.

initialize all AM, BM, and CM registers at the start of calculations for each ring.

Statement number 000280 indicates the number of rings of strain gage rosettes for which data is to be computed. This is indicated by the number of times the "DO LOOP" is used.

PART 2, COMPUTATION OF BEST SLOPE

Part 2 of the program reads the actual strain gage readings in micro-inches per inch from data cards at the end of the program and by the method of least squares determines the slope of the line connecting the data points from 0% to 100% of test depth. The slope is given by the value of B(M) for each gage. The actual strain at zero percent depth is given by the calculated intercept, identified in the program by AZ. The zero percent depth intercept is not used in this program, however.

An indication of the accuracy of the data can be determined in this part of the program. One of the quantities calculated in this part, identified in the program by DIFFS(I), gives the difference between each actual strain gage reading and the calculated best line fitting value at each reading depth. The magnitude of DIFFS(I) will indicate the accuracy of the data.

In future applications DIFFS(I) could be used to expand the program to accept or reject each and every data point by inspecting the magnitude of DIFFS(I) compared to the values of DIFFS(I) for a particular gage at other depths. The program at present does not reject any individual data points. If one or more points were to be rejected, the program could be altered to compute a modified slope and intercept.

The slope, B(M), in this program is given in strain per percent of test depth.

Part 2 of the program is run three times before going on to part 3 of

the program in order to have all three strains available at once when computing the principal stresses for each rosette.

PART 3, COMPUTATION OF PRINCIPAL STRESSES AND AXES

Part three of the program utilizes the strain per percent of test depth calculated in part two of the program for each element of the rosettes to determine the two principal stresses and direction of maximum principal stress taking the direction of flow within the pipe as the zero angle. The values of the two principal stresses are given in pounds per square inch per percent of test depth.

Statements 000860, 000870, and 000880 check to see if any of the individual element strains are zero. If any individual strain is zero, the program jumps to statement 001020 which sets the two principal stresses and angle equal to zero.

Statements 000890 through 001000 calculate two principal stresses using the formulas

$$\sigma_{max} = \frac{E}{2}\left[\frac{\epsilon_1+\epsilon_3}{1-\mu} + \frac{1}{1+\mu}\sqrt{(\epsilon_1-\epsilon_3)^2 + [2\epsilon_2-(\epsilon_1+\epsilon_3)]^2}\ \right]$$

$$\sigma_{min} = \frac{E}{2}\left[\frac{\epsilon_1+\epsilon_3}{1-\mu} - \frac{1}{1+\mu}\sqrt{(\epsilon_1-\epsilon_3)^2 + [2\epsilon_2-(\epsilon_1+\epsilon_3)]^2}\ \right]$$

and the angle using the formula

$$\text{Angle} = \frac{1}{2}\arctan\frac{2\epsilon_2-(\epsilon_1+\epsilon_3)}{\epsilon_1-\epsilon_3}$$

Part 2 and part 3 of the program up to statement 001050 is run twelve times before continuing further. The number of times these two parts are run is determined by the number of rosettes per rosette ring. For the data evaluated, there were twelve rosettes per ring.

Statements 001060 through 001090 orient the angle of the maximum principal stress with respect to the direction of flow through the pipe. This is only necessary when one of the outside legs of the rosette is not oriented in the direction of flow. The rosette orientation as indicated

in figure 1.2.1 has caused the necessity in this particular case for including statements 001060 through 001090.

A print out of values of σ_{MAX}, σ_{MIN} and θ_P for all twelve rosettes in a ring is called for in statements 001100 and 001110.

PART 4, DECISION REGARDING METHOD OF SOLUTION

Part four of the program determines whether the short or long solution to solve for the smoothed values of σ_N and τ_{XY} must be used.

In subsequent sections of this thesis the use of the word inoperative may have one of two meanings. In one meaning, it is used to represent any rosette element for which data was not recorded[5]. In the other meaning, it is used to indicate any rosette element with a zero slope as determined by the method of least squares. The second meaning was a decision made by the writer based on a close evaluation of the data without its use.

When any element of a rosette is inoperative, the value of σ_{MAX}, calculated in part three, for that rosette is set equal to zero. This part of the program checks to see if any value of σ_{MAX} for a ring is zero. If any one value of σ_{MAX} is zero, the program jumps to the long method of solution. If none of the values of σ_{MAX} for a ring is zero, the program goes to the short method of solution.

PART 5, DETERMINATION OF FOURIER COEFFICIENTS - SHORT METHOD

Part five of the program determines the values of the coefficients used in the determination of the smoothed values of σ_N and τ_{XY} by means of the short method as defined in section 2 of this thesis. The computer space names are used in both parts five and six in determining the same items.

[5]When data has not been recorded for a rosette element, the strain readings entered on the program data card for that element are zero. This is done in order to keep proper data card sequence in the program.

Statement 001230 changes the angle Θ_p from degrees to radians while statements 001240 through 001260 solve for the experimental values of σ_N and τ_{xy}. Phi, statement number 001280, is the angle of the rosette from rosette number 12. Statements 001290 through 001400 compute the values of the sines, cosines, sines squared, and cosines squared of the single angle phi, twice the angle phi and three times the angle phi. The value of AM(1) in statement 001410 indicates the number of rosettes in each ring while statements 001420 through 001470 sum up the squares of the sines and cosines of the single, double and triple angles, phi. These last seven statements are the main diagonal elements of the 7 x 7 matrix shown in theory (equation 2.4.15). Statements 001480 through 001610 calculate the values of the two 7 x 1 matrices shown in theory (equations 2.4.17 & 2.4.21). Statements 001620 through 001650 change the values of σ_N and τ_{xy} into values needed in making the polar plots. The last number of the "DO LOOP" which connects statements 001220 through 001660 is determined by the number of rosettes in the ring.

Statements 001690 through 001780 are name changing statements.

Statements 001790 through 001850 solve the seven equations for the seven unknown coefficients for the smoothed values of σ_N while statements 001860 through 001920 solve the seven equations for the seven unknown coefficients for the smoothed values of τ_{xy}. These 14 equations are equations 2.4.18 and 2.4.22 in section two covering theory.

The remaining statements in this part are all print and format statements to give a print out of various items computed in this part.

When part five is used in the program, statement 002010 is used to bypass part six of the program and go directly from part five to part seven.

PART 6, DETERMINATION OF FOURIER COEFFICIENTS - LONG METHOD

Part six of the program determines the values of the coefficients used in the determination of the smoothed values of σ_N and τ_{xy} by means of the long method as defined in section 2 of this thesis.

Statement 002080 is used to determine if any rosette in the ring is inoperative. If any rosette is inoperative, statements 002090 through 002150 are used and the program goes to the end of the "DO LOOP". This is used to prevent plotting an experimental value of σ_N and τ_{xy} on the output plots when the rosette is inoperative.

Statements 002160 through 002290 are identical to statements 001230 through 001360 of part five. Statements 002300 through 002480 compute the combinations of sines and cosines of the single, double and triple angle phi. The items identified by AM(1) through AM(28), statements 002490 through 002760, are the various elements of the 7 x 7 matrix shown in section two (equation 2.4.15). Statements 002770 through 002830 are the elements of the 7 x 1 matrix shown in theory (equation 2.4.17) while statements 002840 through 002900 are the elements of the 7 x 1 matrix shown in theory by equation 2.4.21. Statements 002910 through 002940 are identical to statements 001620 through 001650 in part five of the program. The last number of the "DO LOOP" connecting statements 002070 through 002950 is once more determined by the number of rosettes in each ring.

Statements 002980 through 003460 identify all elements of the 7 x 7 matrix individually (Equation 2.4.15) as can be noted by the double subscripts. In turn, statements 003500 through 003520 do the identical identification for the two 7 x 1 matrices (Equations 2.4.17 and 2.4.21). Statements 003530, 003540 and 003550 are the first three items needed in the CALL GAUSS 2 statements. The two CALL GAUSS 2 statements, numbers

003560 and 003610, call the subroutine GAUSS 2 into use. GAUSS 2 solves for the unknowns in the matrix equations and is used twice to solve for the unknown coefficients of both σ_N and τ_{xy}.

The various other statements in this part of the program are print and format statements which print out various quantities computed in this part.

PART 7, PREPARATION FOR GRAPHICAL OUTPUT

Part seven of the program is directly related to the subroutine DRAW. It is merely a series of statements which set up the labels placed on figures drawn by the program. More information on the titles and the requirement for them can be obtained by consulting Appendix D. Statements 003750 through 003900 are a computed GO TO series for numbering the rings appropriately as computer draws the figures.

PART 8, COMPUTATION OF SMOOTHED VALUES OF AXIAL AND SHEARING STRESS

Part eight of the program computes the smoothed values of σ_N and τ_{xy} around the entire pipe diameter and converts the values into polar coordinates for plotting purposes. The 360 degrees of the circle of the pipe circumference is divided into 628 segments for plotting purposes. Since the computer plotter used for figure drawing can only draw vertical, horizontal and forty-five degree lines, it was necessary to divide the pipe circumference into as many convenient increments as possible. The value 628 was chosen since the computer utilizes radians, not degrees for angles, and 628 is a convenient number with which to divide 2π radians and 900 is the limiting number of points that can be plotted by the subroutine DRAW.

Statement 003970 divides 2π by 628. Statement 004000 computes the angle. Statements 004010 through 004060 compute the sines and cosines of the angles needed in the formulas for σ_N and τ_{xy}. Statements 004070

and 004080 compute the smoothed values of σ_N using the formula

$$\sigma_N = a + b\sin\phi + c\cos\phi + d\sin 2\phi + e\cos 2\phi + f\sin 3\phi + g\cos 3\phi .$$

Statements 004090 and 004100 compute the smoothed value of τ_{xy} using the formula

$$\tau_{xy} = k + l\sin\phi + m\cos\phi + n\sin 2\phi + o\cos 2\phi + p\sin 3\phi + q\cos 3\phi .$$

Statements 004110 through 004180 transform the values of σ_N and τ_{xy} into values that can be utilized in polar plotting. Statements 004110, 004120, 004150, and 004160 are used for drawing reference circles on the figures. These reference circles indicate the zero stress ring. Any reading inside the zero stress circle for σ_N can be taken as a compression, while any reading outside the zero stress circle for σ_N can be taken as a tension. For the plots of τ_{xy}, any reading inside the zero stress circle can be taken as a negative shear, while any reading outside the zero stress circle can be taken as a positive shear. A positive shear is defined as a shear caused by the portion downstream of the rosette ring acting in the direction of increasing rosette numbers on the upstream portion. A negative shear is defined as a shear caused by the portion downstream of a rosette ring acting in the direction of decreasing rosette numbers on the upstream portion. The number 100, chosen for plotting the zero stress circle for σ_N and 40, chosen for plotting the zero stress circle for τ_{xy}, were chosen to keep all plotted values positive and to show as much clarity on the figures as possible. If, in future applications of this program, these scales are not convenient, change the numbers 100 and 40 in statements 001620-001650 inclusive, 002910-002940 inclusive and 004110-004180 inclusive to more appropriate values.

PART 9, GRAPHICAL OUTPUT

Part nine of the program consists of three statements which call the Subroutine DRAW for plotting the experimental and smoothed values of

σ_N around the circumference of each pipe section at a particular rosette, a title change statement, and three more statements which call the Subroutine DRAW for similar plotting of τ_{XY} values. The CALL DRAW statements occur in groups of three; the first one plots a circle of zero stress, the second one plots the smoothed points, and the third one plots the experimental values. Since compression stress appears as a negative stress while tension stress appears as a positive stress, a polar plot could not be utilized unless the position of zero stress was far enough away from the center of the plot to prevent the stresses from being plotted 180 degrees out of position. The proper scaling of the plots must be done before the program is utilized. Appendix D , the Subroutine DRAW writeup, gives full details on both scaling and what should be placed in a CALL DRAW statement.

PART 10, COMPUTATION OF FORCE AND MOMENT COMPONENTS

Part ten of the program computes components of the three forces and the three moments at each ring which are used for the equilibrium check. The x-direction is taken as the direction of flow in the pipe. The y-direction is taken as the direction from the center of the pipe toward rosette number 12. The z-direction is taken as the direction from the center of the pipe toward rosette number three. XFOR is the force in the x-direction and XMOM is the moment in the x-direction. YFOR is the force in the y-direction and YMOM is the moment in the y-direction. ZFOR is the force in the z-direction and ZMOM is the moment in the z-direction. The three force components and three moment components are used in the manual computation of the three equilibrium forces and three equilibrium moments.

PART 11, GAUSS 2, SOLUTION OF SIMULTANEOUS EQUATIONS

Part eleven of the program is the Subroutine GAUSS 2. This is a standard subroutine in the computer subroutine library at the U. S. Naval

Postgraduate School. The computer facility explanation of this subroutine is Appendix C of this thesis.

PART 12, DRAW - GENERAL PROGRAM FOR GRAPHICAL OUTPUT

Part twelve of the program is the Subroutine DRAW. This is also a standard subroutine in the computer facility subroutine library at the U. S. Naval Postgraduate School. The computer facility explanation of this subroutine is Appendix D of this thesis.

4. Data Evaluation

GAGE ORIENTATION

The gage orientation is one of the important items to which careful thought should be given in order to obtain the most meaningful data. The most important consideration concerning gage orientation is the number of rosettes per ring and the spacing between rosettes. As previously discussed in section 2, it is easier to obtain smoothed values of σ_N and τ_{xy} when an even number of rosettes equally spaced are employed because equations with only one unknown are more readily solved than several equations with several unknowns. The actual number of rosettes per ring employed in the SSBN 640 instrumentation appears to be a good number for the size piping system to be analyzed. On future instrumentation projects the size of the piping system should be considered in determining the even number of gage rosettes to be used.

In the system analyzed the rosette orientation was shifted 45° alternately between even and odd numbered gages (See figure 1.2.1). This shift in orientation caused minor difficulties in obtaining the proper angle between the maximum principal stress and direction of flow which had to be checked very closely; it is also a source of possible misunderstanding. The identical results could have been obtained with all rosettes oriented identically and thereby eliminating these minor difficulties. It is highly recommended for future instrumentations that all rosettes be oriented identically.

It might be beneficial to consider the use of T- DELTA rosettes for future instrumentation projects. In several rosettes, one leg indicated erroneous readings. If a fourth leg of the rosette had been available, the rosette might have produced meaningful results not otherwise obtainable.

ACCURACY OF READINGS - DOCK TRIAL DATA

The readings taken during dockside hydrostatic tests were recorded to the nearest 10 μin/in. Since it is possible to read an SR-4 Indicator, Type N, more accurately than the nearest 10 μin/in, the accuracy of all readings is not considered to be the best possible. It is desirable in future tests where an SR-4 Indicator, Type N, is used to read or estimate strain gage indications to 1 μin/in. Seven pressure increments were recorded during the tests.

One of the items printed out in Part 2 of the digital computer program is DIFFS(I), the difference between each actual strain gage reading and the least squares calculated best line fitting value. In evaluating the actual data, any value of DIFFS(I) greater than 10 μin/in is considered to indicate an inaccurate reading. With this as a basis of inaccuracy, the following statistical tabulation of the data has been made.

Table 1

Total number of readings taken		1939
Total number of readings with DIFFS(I) greater than 10 μin/in		49
% of total in error		2.52%
Total number of gage elements		288
Number of inoperative elements		11
Number of elements with one reading more than 10 μin/in in error		15
Number of elements with two readings more than 10 μin/in in error		9
Number of elements with three readings more than 10 μin/in in error		4
Number of elements with four readings more than 10 μin/in in error		1
Total number of inaccurate or inoperative elements		40
% of total elements that were inaccurate or inoperative		13.89%
Total number of rosettes		96
Number of rosettes with at least one inaccurate reading or an inoperative element	Ring 1	4
	Ring 2	3
	Ring 3	4
	Ring 4	3
	Ring 5	3
	Ring 6	7
	Ring 7	2
	Ring 8	4
	Total	30
% of total rosettes inaccurate or inoperative		31.25%
Total number of rosettes inoperative due to faulty element		9

% of total rosettes inoperative due to faulty element 9.375%
Ring 6 had one rosette with all three elements faulty

The statistical percentages shown in Table 1 have been interpreted in the following manner:

a. Faulty elements caused 9.375% of the rosettes to be unusable in evaluating the system. Although this percentage is considered to be high in obtaining good experimental results, it is not excessive in evaluating this system.

b. The percentage of inaccurate rosettes caused by a reading error of 10 μin/in, 21.875%, does not have any real meaning because only one, two, or even three inaccurate readings might not have caused the slope of the strain depth curve to be enough in error to make the rosette give a faulty overall stress calculation.

c. The percentage of inaccurate readings, 2.52%, is well within the experimental accuracy expected, if noteven less than could be expected from a test performed under the conditions of this test.

PRINCIPAL STRESS RESULTS - DOCKSIDE HYDROSTATIC TESTS

The Bureau of Ships Shipbuilding Specification, MILT 16420G, states that the maximum allowable stress in 70-30 copper nickel piping under cyclic loads is 15000 psi. In publications on material properties, the yield strength of 70-30 copper nickel is 25,000 psi. The maximum value of σ_{max} at any rosette at 100% test depth calculated from the dockside hydrostatic test data was found to be slightly less than 10,000 psi. This indicates that the maximum principal stress does not reach the allowable stress and is well below the yield strength at any instrumented point. Since the rosette ring locations were chosen to be representative of the entire system, this is sufficient proof that in all probability the stresses within the piping system do not exceed the allowable stress limit of the material. It is also proof that the stress levels were

sufficiently low to assure elastic action, as is assumed in the analysis of the data.

COMPARISON OF EXPERIMENTAL DATA TO SMOOTHED CURVES

The sixteen figures appearing in Appendix E are polar plots of smoothed axial and shearing stress around the circumference of the piping at each rosette ring and the experimental values of the axial and shearing stress at the individual rosette locations. The circle ten units in radius on the σ_N plots and four units in radius on τ_{xy} plots is the circle of zero stress on these plots. For the σ_N plots stresses inside the zero stress circle are compression while those outside the circles are tension. For the τ_{xy} plots stress inside the zero stress circle is shearing stress in the negative direction while stress outside the zero stress circle is positive shearing stress. The scales on the plots are arbitrary to remain within the unclassified security classification as noted in section 1. The stresses shown are plotted per percent of test. depth.

It is readily apparent by close examination of the sixteen figures that the experimental values are in close agreement with the smoothed curves. (The experimental points are marked by an x while the smoothed values appear as a closed curve.) The x at the center of some plots is to be ignored as it is only a necessary step in having the CDC 1604 computer draw the plots. It will also be noticed that some plots do not have all twelve experimental points shown. The missing experimental points are an indication of the location of faulty rosettes (inoperative elements) as no data from fully operative rosettes was rejected in this evaluation. Had the missing rosettes been operative, the shape of the curves might have been altered.

EQUILIBRIUM RESULTS FROM DOCK-TRIAL DATA

Equilibrium calculations have been performed on the segments of piping between rosette rings 1 and 2, 3 and 4, and 7 and 8. Calculations were not performed on segments connecting any other two rings because unknown forces due to foundations, pipe hangers, or other constraints exist between these segments. Table 2 gives a tabulation on the three segments analyzed.

Table 2

SEGMENT	Fx	Fy	Fz	Mx	My	Mz	Fx	Fy	Fz	Mx	My	Mz	Frx	Fry	Frz	Mrx	Mry	Mrz
1 - 2	At end #1						At end #2						RESULTANT					
	-13	-53	6	447	215	66	-13	-81	-6	-176	116	-659	-26	-134	0	1259	30	-812
3 - 4	At end #3						At end #4						RESULTANT					
	-10	-14	97	883	-740	-486	18	17	15	511	621	-518	8	3	112	1190	227	1208
7 - 8	At end #7						At end #8						RESULTANT					
	-52	12	20	-265	-176	56	-7	8	-88	-255	274	122	-59	20	-68	-510	1294	327

(Resultant moments are about lower numbered end of segments)

Theoretically, for equilibrium to be satisfied, all values of F_{Rx}, F_{Ry}, F_{Rz}, M_{Rx}, M_{Ry}, and M_{Rz} should be zero. Although only one of the eighteen resultant values is zero, the other values are not excessively high. Any one of the following influences could have caused the non-zero values.

a. The distances between rosette rings that were used in calculating the resultant moments were distances measured from the piping blueprints using the approximate locations of the rosette rings. Since exact values of these distances were not available, these were the best values available to use even though they could be a source of error in the calculations.

b. The values of pipe radius and thickness used in calculating the forces and moments at each rosette ring were average values calculated by averaging the values of thickness and radius at each rosette within the

ring. In addition the values of radius at each rosette reported in Propulsion Plant Test Form 721-5 were obtained by measuring pipe diameters and dividing by two, which is not a true value of the radius considering that the circumference of the pipe is not a true circle. Since outside diameter measurements for rosette ring number 1 were not recorded, the value of radius of ring 1 used in force and moment calculations was the average of the other seven rings.

c. The value of the forces exerted by the fluid within the pipe on the inside surface of the pipe was determined using average values of radius and wall thickness. Using average values instead of exact values could have once more caused errors.

d. Errors in the Fourier coefficients caused by errors in strain gage readings could also have contributed to errors in the forces and moments calculated at each rosette ring.

Considering all the possible sources of error in the equilibrium calculations, it is reasonable to expect that all the resultant forces and moments would not be zero. However, some of the values in table 2 are much larger than could be considered reasonable for the method used.

5. Conclusion

VALIDITY OF RESULTS

As indicated by the results shown in this thesis, the method developed can readily be used to obtain stress information concerning the system. However, the validity of the equilibrium results will depend largely on the accuracy of the strain gage data and other data needed for the equilibrium calculations.

In order to increase the accuracy of the data and thereby increase the validity of the results, the following recommendations are submitted:

a. Appoint a project engineer with experience in piping theory and strain gage measurements to have overall control of the instrumentation of the system, data taking during all tests, and reporting of results obtained from the data. This one person would supervise the project from the start of instrumentation to the submission of the final report. Having one man in charge of the project from start to finish will greatly reduce the chances of faulty communication between people working on the project, reduce the chances of necessary data items being omitted from the data taken, and reduce the chances for error in the actual data taken.

b. Utilize automatic scanning and recording equipment to record all strain gage data during actual tests. Automatic recording equipment can observe strain gage reading much more accurately than equipment read manually and in much less time. In addition, human recording errors could be eliminated. A rapid record of strain gage reading at any depth could also be taken using the automatic scanning and recording equipment.

c. An important aid to the person analyzing the experimental data from these tests would be notes taken during the period that data is being taken. Some of the information that would be useful as notes is any unusual conditions that occurred during data taking, comments

concerning any data that might give an immediate indication as to being inaccurate or unreliable, the conditions that existed during the periods between the taking of data, etc.. The project engineer should be observing all tests to take his own notes as well as having notes taken by all technicians involved in the tests.

d. Additional data to be taken during the tests should include actual measured distances between two adjacent rosette rings for use in more accurate equilibrium calculations and the location of the welded seams for each section of pipe.

By further refinement of data taking methods and actual data taken, the method of evaluation developed in this thesis can be used as a tool in analyzing any piping system instrumented in a manner similiar to the Main Seawater Piping System on SSBN 640.

SUGGESTED FUTURE REFINEMENTS FOR METHOD OF EVALUATION

Certain refinements to the present method of evaluation have been considered; however, sufficient time is not available to permit these ideas to be included in this thesis. The basic content of these ideas is listed in the following paragraphs.

Part 2 of the program could be expanded to include a method which would investigate the value of DIFFS(I). This would be a two stage process. The zero percent depth intercept, a; slope, b; and values of DIFFS(I) would be found as they are in the present program. If DIFFS(I) were greater than a predetermined limit value, the data which produced that DIFFS(I) value would be rejected and new values of zero percent depth intercept and slope would be found. An addition of the type proposed would improve the accuracy of all resultant calculations.

The present theory might be further developed to find a method to make some use of strain gage rosettes which have one or two inoperative

elements. When the center element is the only one inoperative, some useful information may obviously be obtained from the rosette. If the orientation is that of the even numbered rosettes as shown in figure 1.2.1, elements A and C operative and B inoperative would permit evaluating the axial stress. If the orientation is that of the odd numbered rosettes as shown in figure 1.2.1, elements B and C operative and A inoperative would permit evaluating a shear stress. Beyond this, if rosette failure were other than as indicated above, there is probably some statistical way of making use of the data afforded by the operative elements. However, the theory of such usage remains to be worked out.

These two refinements added to the analysis method might produce more meaningful results from the dockside hydrostatic test data.

Bibliography

1. Beckwith, T.G. and Buck, N.L., Mechanical Measurements, Addison-Wesley Publishing Company, Inc., 1961.

2. Dove, R.C. and Adams, P.H., Experimental Stress Analysis and Motion Measurement, Charles E. Merrill Books, Inc., 1964.

3. Lee, G.H., An Introduction to Experimental Stress Analysis, John Wiley and Sons, Inc., 1950.

4. Leland, O.M., Practical Least Squares, McGraw-Hill Book Company, Inc., 1921.

5. Perry, C.C. and Lissner, H.R., The Strain Gage Primer, McGraw-Hill Book Company, Inc., 1962.

6. Spiegel, M.R., Theory and Problems of Statistics, Schaum Publishing Company, 1961.

7. General Dynamics Corporation/Electric Boat Division, Propulsion Plant Test Form Number 721-5 Rev. A, 1965.

8. General Dynamics Corporation/Electric Boat Division, Sea Trial Form Number 21.04 Rev A, 1965.

APPENDIX A

COMPLETE PROGRAM LISTING

```
..JOB0433E,  KOCH, J W
      PROGRAM STRAIN
      DIMENSION X(8),XSAVE(8),Y(8),YSAVE(8),YFIT(8),DIFFS(8),B(3),AM(22)
     1,BM(11),CM(11),A(11,11),BN(11,1),CN(11,1),ZX(11,1),XY(11,1),PHI(12
     2),PHI1(630),SIGXX(12),TAUXY(12),ITITLE(12),SIGN7(630),TAUN7(630),S
     3MAX(12),SMIN(12),ANG(12),THETA(12),XSXX(12),YSXX(12),XTXY(12),YTXY
     4(12),XSNR(630),YSNR(630),XSN7(630),YSN7(630),XTNR(630),YTNR(630),X
     5TN7(630),YTN7(630),TH(8),RA(8),XFOR(8),YFOR(8),ZFOR(8),XMOM(8),YMO
     6M(8),ZMOM(8)
C  THIS IS THE START OF PART 1.  PART 1 READS IN DATA.
      READ 999, N,E,U
  999 FORMAT(1I4,2F15.5)
      TH(1)=0.625
      RA(1)=6.40
      TH(2)=0.634
      RA(2)=6.390
      TH(3)=0.620
      RA(3)=6.409
      TH(4)=0.633
      RA(4)=6.374
      TH(5)=0.428
      RA(5)=6.416
      TH(6)=0.640
      RA(6)=6.354
      TH(7)=0.423
      RA(7)=6.411
      TH(8)=0.423
      RA(8)=6.410
      DO 85 K=1,8
      DO 3 I=1,28
    3 AM(I)=0.
      DO 4 I=1,7
      BM(I)=0.
    4 CM(I)=0.
      Z=3.14159/6.
C  THIS IS THE END OF PART 1.
   13 DO 15 L=1,12
C  THIS IS THE START OF PART 2.  PART 2 READS THE DATA AT THE END OF THE
C  PROGRAM TO DETERMINE THE STRAIN PER PERCENT OF DEPTH FOR EACH
C  INDIVIDUAL GAGE.
   57 DO 58 M= 1,3
      READ 100, (X(I),Y(I),I=1,N)
  100 FORMAT(14F5.0)
      DO 996 I=1,N
      XSAVE(I)=X(I)
  996 YSAVE(I)=Y(I)
      YSUM = 0.
      SUMX = 0.
      SUMY = 0.
      SUMXY = 0.
      SUMXX = 0.
      XN = N
      DO 30 I = 1, N
      SUMX=SUMX+X(I)
      SUMY = SUMY+Y(I)
      SUMXX = SUMXX + (X(I)*X(I))
```

```
   30 SUMXY=SUMXY+(X(I)*Y(I))                                          000560
      DENOM = XN*SUMXX-SUMX*SUMX                                       000570
      ANUM = SUMY * SUMXX - SUMX * SUMXY                               000580
      AZ = ANUM / DENOM                                                000590
      BNUM= XN*SUMXY-SUMX*SUMY                                         000600
      B(M) = BNUM / DENOM                                              000610
      DO 40 J = 1,N                                                    000620
      YC = AZ +B(M) *X(J)                                              000630
      DIFF = Y(J) - YC                                                 000640
   40 YSUM = YSUM + DIFF*DIFF                                          000650
      S2 = YSUM / (XN - 1.)                                            000660
      SIG=S2/DENOM                                                     000670
      SRA=SQRTF(SUMXX*SIG)                                             000680
      SRB=SQRTF(XN*SIG)                                                000690
      PRINT 997, AZ, B(M), SRA, SRB                                    000700
  997 FORMAT(24H0  STRAIGHT LINE FITTING// 25H USING THE FORM Y=BX + A,  000710
     1/ 4H A = E14.8/ 7H B(M) = E14.8/ 26H STANDARD DEVIATION OF A = E14 000720
     2.8/ 29H STANDARD DEVIATION OF B(M) = E14.8)                      000730
  801 SUMFIT=.0                                                        000740
      DO 90 I=1,N                                                      000750
      YFIT(I) = B(M)*X(I)+AZ                                           000760
      DIFFS(I)=YFIT(I)-Y(I)                                            000770
   90 SUMFIT=SUMFIT+ DIFFS(I)                                          000780
  995 PRINT 99,SUMFIT,(X(I),YSAVE(I),Y(I),YFIT(I),DIFFS(I),I=1,N)      000790
   99 FORMAT(9H0SUMFIT =E12.5///10X1HX14X1HY9X10HYTRANSPOSE8X4HYFIT11X5H 000800
     1DIFFS///(5E15.5))                                                000810
   58 CONTINUE                                                         000820
C  THIS IS THE END OF PART 2 AND THE START OF PART 3.  THIS PART       000830
C  DETERMINES THE PRINCIPAL STRESSES AND THE ANGLE FROM THE PIPE       000840
C  CENTERLINE OF THE MAXIMUM PRINCIPAL STRESS FOR EACH ROSETTE.        000850
      IF(B(1)) 401,410,401                                             000860
  401 IF(B(2)) 402,410,402                                             000870
  402 IF(B(3)) 403,410,403                                             000880
  403 AA=E/2.                                                          000890
      BA=B(1) + B(3)                                                   000900
      CA=B(1) - B(3)                                                   000910
      D=1.+U                                                           000920
      F=1.-U                                                           000930
      G=2.*B(2) - BA                                                   000940
      H=BA/F                                                           000950
      P=1./D*SQRTF(CA*CA+G*G)                                          000960
      SMAX(L)=AA*(H+P)                                                 000970
      SMIN(L)=AA*(H-P)                                                 000980
      ANG(L)=0.5*(ATANF(G/CA)+(3.14159/2.)*(1.-(CA/ABSF(CA))))*180./3.14 000990
     1159                                                              001000
      GO TO 15                                                         001010
  410 SMAX(L)=0.                                                       001020
      SMIN(L)=0.                                                       001030
      ANG(L)=0.                                                        001040
   15 CONTINUE                                                         001050
      DO 115 L=1,11,2                                                  001060
  115 THETA(L)=ANG(L)-45.                                              001070
      DO 116 L=2,12,2                                                  001080
  116 THETA(L)=ANG(L)                                                  001090
      PRINT 5,(SMAX(L),SMIN(L),THETA(L),L=1,12)                        001100
    5 FORMAT(///6X7HSMAX(L)8X7HSMIN(L)8X8HTHETA(L)///(3E15.5))         001110
```

```
C  THIS IS THE END OF PART 3 AND THE START OF PART 4.  PART 4 DETERMINES  001120
C  WHETHER THE SHORT OR LONG SOLUTION TO SOLVE THE STRESS DISTRIBUTION    001130
C  MUST BE USED.                                                         001140
      DO 420 L=1,12                                                      001150
      IF(SMAX(L)) 420,430,420                                            001160
  420 CONTINUE                                                           001170
C  THIS IS THE END OF PART 4 AND THE START OF PART 5.  PART 5 DETERMINES  001180
C  THE VALUES OF THE COEFFICIENTS USED IN THE THEORETICAL DETERMINATION   001190
C  OF NORMAL STRESS AND SHEAR STRESS BY THE SHORT OR EVEN NUMBER OF       001200
C  GAGES EQUALLY SPACED METHOD.                                          001210
      DO 16 L=1,12                                                       001220
      THETA(L)=THETA(L)*3.14159/180.                                     001230
      SIGXX(L)=SMAX(L)*COSF(THETA(L))*COSF(THETA(L))+SMIN(L)*SINF(THETA(  001240
     1L))*SINF(THETA(L))                                                 001250
      TAUXY(L)=(SMAX(L)-SMIN(L))*0.5*SINF(2.*THETA(L))                   001260
      XL=L                                                               001270
      PHI(L)=XL*Z                                                        001280
      SI=SINF(PHI(L))                                                    001290
      CI=COSF(PHI(L))                                                    001300
      SI2=SINF(2.*PHI(L))                                                001310
      CI2=COSF(2.*PHI(L))                                                001320
      SI3=SINF(3.*PHI(L))                                                001330
      CI3=COSF(3.*PHI(L))                                                001340
      SISQ=SI*SI                                                         001350
      CISQ=CI*CI                                                         001360
      S2SQ=SI2*SI2                                                       001370
      C2SQ=CI2*CI2                                                       001380
      S3SQ=SI3*SI3                                                       001390
      C3SQ=CI3*CI3                                                       001400
      AM(1)=12.                                                          001410
      AM(6)=AM(6)+SISQ                                                   001420
      AM(10)=AM(10)+CISQ                                                 001430
      AM(13)=AM(13)+S2SQ                                                 001440
      AM(15)=AM(15)+C2SQ                                                 001450
      AM(26)=AM(26)+S3SQ                                                 001460
      AM(28)=AM(28)+C3SQ                                                 001470
      BM(1)=BM(1)+SIGXX(L)                                               001480
      BM(2)=BM(2)+SI*SIGXX(L)                                            001490
      BM(3)=BM(3)+CI*SIGXX(L)                                            001500
      BM(4)=BM(4)+SI2*SIGXX(L)                                           001510
      BM(5)=BM(5)+CI2*SIGXX(L)                                           001520
      BM(6)=BM(6)+SI3*SIGXX(L)                                           001530
      BM(7)=BM(7)+CI3*SIGXX(L)                                           001540
      CM(1)=CM(1)+TAUXY(L)                                               001550
      CM(2)=CM(2)+SI*TAUXY(L)                                            001560
      CM(3)=CM(3)+CI*TAUXY(L)                                            001570
      CM(4)=CM(4)+SI2*TAUXY(L)                                           001580
      CM(5)=CM(5)+CI2*TAUXY(L)                                           001590
      CM(6)=CM(6)+SI3*TAUXY(L)                                           001600
      CM(7)=CM(7)+CI3*TAUXY(L)                                           001610
      XSXX(L)=(100.+SIGXX(L))*CI                                         001620
      YSXX(L)=(100.+SIGXX(L))*SI                                         001630
      XTXY(L)=(40.+TAUXY(L))*CI                                          001640
      YTXY(L)=(40.+TAUXY(L))*SI                                          001650
   16 CONTINUE                                                           001660
      PRINT 7,(SIGXX(L),TAUXY(L),L=1,12)                                 001670
```

```
    7 FORMAT(///6X5HSIGXX1PX6HTAUXY////(2E13.5))                          001680
      A(1,1)=AM(1)                                                        001690
      A(2,2)=AM(6)                                                        001700
      A(3,3)=AM(10)                                                       001710
      A(4,4)=AM(13)                                                       001720
      A(5,5)=AM(15)                                                       001730
      A(6,6)=AM(26)                                                       001740
      A(7,7)=AM(28)                                                       001750
      DO 11 I=1,7                                                         001760
      BN(I,1)=BM(I)                                                       001770
   11 CN(I,1)=CM(I)                                                       001780
      ZX(1,1)=BN(1,1)/A(1,1)                                              001790
      ZX(2,1)=BN(2,1)/A(2,2)                                              001800
      ZX(3,1)=BN(3,1)/A(3,3)                                              001810
      ZX(4,1)=BN(4,1)/A(4,4)                                              001820
      ZX(5,1)=BN(5,1)/A(5,5)                                              001830
      ZX(6,1)=BN(6,1)/A(6,6)                                              001840
      ZX(7,1)=BN(7,1)/A(7,7)                                              001850
      XY(1,1)=CN(1,1)/A(1,1)                                              001860
      XY(2,1)=CN(2,1)/A(2,2)                                              001870
      XY(3,1)=CN(3,1)/A(3,3)                                              001880
      XY(4,1)=CN(4,1)/A(4,4)                                              001890
      XY(5,1)=CN(5,1)/A(5,5)                                              001900
      XY(6,1)=CN(6,1)/A(6,6)                                              001910
      XY(7,1)=CN(7,1)/A(7,7)                                              001920
      PRINT 196,ZX(1,1),ZX(2,1),ZX(3,1),ZX(4,1),ZX(5,1),ZX(6,1),ZX(7,1)   001930
  196 FORMAT(///11H ZX(1,1) = E14.8//11H ZX(2,1) = E14.8//11H ZX(3,1) =   001940
     1E14.8//11H ZX(4,1) = E14.8//11H ZX(5,1) = E14.8//11H ZX(6,1) = E14  001950
     2.8//11H ZX(7,1) = E14.8//)                                          001960
      PRINT 156,XY(1,1),XY(2,1),XY(3,1),XY(4,1),XY(5,1),XY(6,1),XY(7,1)   001970
  156 FORMAT(///11H XY(1,1) = E14.8//11H XY(2,1) = E14.8//11H XY(3,1) =   001980
     1E14.8//11H XY(4,1) = E14.8//11H XY(5,1) = E14.8//11H XY(6,1) = E14  001990
     2.8//11H XY(7,1) = E14.8//)                                          002000
      GO TO 456                                                           002010
C  THIS IS THE END OF PART 5 AND THE START OF PART 6.  PART 6 DETERMINES  002020
C  THE VALUES OF THE COEFFICIENTS USED IN THE THEORETICAL DETERMINATION   002030
C  OF THE VALUES OF THE COEFFICIENTS USED IN THE THEORETICAL              002040
C  DETERMINATION OF NORMAL STRESS AND SHEAR STRESS BY THE LONG OR         002050
C  UNEQUALLY SPACED GAGES METHOD.                                         002060
  430 DO 46 L=1,12                                                        002070
      IF(SMAX(L)) 440,42,440                                              002080
   42 SIGXX(L)=0.                                                         002090
      TAUXY(L)=0.                                                         002100
      XSXX(L)=0.                                                          002110
      YSXX(L)=0.                                                          002120
      XTXY(L)=0.                                                          002130
      YTXY(L)=0.                                                          002140
      GO TO 46                                                            002150
  440 THETA(L)=THETA(L)*3.14159/180.                                      002160
      SIGXX(L)=SMAX(L)*COSF(THETA(L))*COSF(THETA(L))+SMIN(L)*SINF(THETA(  002170
     1L))*SINF(THETA(L))                                                  002180
      TAUXY(L)=(SMAX(L)-SMIN(L))*0.5*SINF(2.*THETA(L))                    002190
      XL=L                                                                002200
      PHI(L)=XL*Z                                                         002210
      SI=SINF(PHI(L))                                                     002220
      CI=COSF(PHI(L))                                                     002230
```

```
SI2=SIN(2.*PHI(L))                                                  002240
CI2=COS(2.*PHI(L))                                                  002250
SI3=SIN(3.*PHI(L))                                                  002260
CI3=COS(3.*PHI(L))                                                  002270
SISQ=SI*SI                                                          002280
CISQ=CI*CI                                                          002290
SC=SI*CI                                                            002300
SIS2=SI*SI2                                                         002310
SIC2=SI*CI2                                                         002320
CIS2=CI*SI2                                                         002330
CIC2=CI*CI2                                                         002340
S2SQ=SI2*SI2                                                        002350
S2C2=SI2*CI2                                                        002360
C2SQ=CI2*CI2                                                        002370
SIS3=SI*SI3                                                         002380
SIC3=SI*CI3                                                         002390
CIS3=CI*SI3                                                         002400
CIC3=CI*CI3                                                         002410
S2S3=SI2*SI3                                                        002420
S2C3=SI2*CI3                                                        002430
C2S3=CI2*SI3                                                        002440
C2C3=CI2*CI3                                                        002450
S3SQ=SI3*SI3                                                        002460
S3C3=SI3*CI3                                                        002470
C3SQ=CI3*CI3                                                        002480
AM(1)=AM(1)+1.                                                      002490
AM(2)=AM(2)+SI                                                      002500
AM(3)=AM(3)+CI                                                      002510
AM(4)=AM(4)+SI2                                                     002520
AM(5)=AM(5)+CI2                                                     002530
AM(6)=AM(6)+SISQ                                                    002540
AM(7)=AM(7)+SC                                                      002550
AM(8)=AM(8)+SIS2                                                    002560
AM(9)=AM(9)+SIC2                                                    002570
AM(10)=AM(10)+CISQ                                                  002580
AM(11)=AM(11)+CIS2                                                  002590
AM(12)=AM(12)+CIC2                                                  002600
AM(13)=AM(13)+S2SQ                                                  002610
AM(14)=AM(14)+S2C2                                                  002620
AM(15)=AM(15)+C2SQ                                                  002630
AM(16)=AM(16)+SI3                                                   002640
AM(17)=AM(17)+CI3                                                   002650
AM(18)=AM(18)+SIS3                                                  002660
AM(19)=AM(19)+SIC3                                                  002670
AM(20)=AM(20)+CIS3                                                  002680
AM(21)=AM(21)+CIC3                                                  002690
AM(22)=AM(22)+S2S3                                                  002700
AM(23)=AM(23)+S2C3                                                  002710
AM(24)=AM(24)+C2S3                                                  002720
AM(25)=AM(25)+C2C3                                                  002730
AM(26)=AM(26)+S3SQ                                                  002740
AM(27)=AM(27)+S3C3                                                  002750
AM(28)=AM(28)+C3SQ                                                  002760
BM(1)=BM(1)+SIGXX(L)                                                002770
BM(2)=BM(2)+SI*SIGXX(L)                                             002780
BM(3)=BM(3)+CI*SIGXX(L)                                             002790
```

```
      BM(4)=BM(4)+SI2*SIGXX(L)                                          002800
      BM(5)=BM(5)+CI2*SIGXX(L)                                          002810
      BM(6)=BM(6)+SI3*SIGXX(L)                                          002820
      BM(7)=BM(7)+CI3*SIGXX(L)                                          002830
      CM(1)=CM(1)+TAUXY(L)                                              002840
      CM(2)=CM(2)+SI*TAUXY(L)                                           002850
      CM(3)=CM(3)+CI*TAUXY(L)                                           002860
      CM(4)=CM(4)+SI2*TAUXY(L)                                          002870
      CM(5)=CM(5)+CI2*TAUXY(L)                                          002880
      CM(6)=CM(6)+SI3*TAUXY(L)                                          002890
      CM(7)=CM(7)+CI3*TAUXY(L)                                          002900
      XSXX(L)=(100.+SIGXX(L))*CI                                        002910
      YSXX(L)=(100.+SIGXX(L))*SI                                        002920
      XTXY(L)=(40.+TAUXY(L))*CI                                         002930
      YTXY(L)=(40.+TAUXY(L))*SI                                         002940
   46 CONTINUE                                                          002950
      PRINT 77,(SIGXX(L),TAUXY(L),L=1,12)                               002960
   77 FORMAT(///6X5HSIGXX10X5HTAUXY///(2E15.5))                         002970
      A(1,1)=AM(1)                                                      002980
      A(1,2)=AM(2)                                                      002990
      A(1,3)=AM(3)                                                      003000
      A(1,4)=AM(4)                                                      003010
      A(1,5)=AM(5)                                                      003020
      A(1,6)=AM(16)                                                     003030
      A(1,7)=AM(17)                                                     003040
      A(2,1)=AM(2)                                                      003050
      A(2,2)=AM(6)                                                      003060
      A(2,3)=AM(7)                                                      003070
      A(2,4)=AM(8)                                                      003080
      A(2,5)=AM(9)                                                      003090
      A(2,6)=AM(18)                                                     003100
      A(2,7)=AM(19)                                                     003110
      A(3,1)=AM(3)                                                      003120
      A(3,2)=AM(7)                                                      003130
      A(3,3)=AM(10)                                                     003140
      A(3,4)=AM(11)                                                     003150
      A(3,5)=AM(12)                                                     003160
      A(3,6)=AM(20)                                                     003170
      A(3,7)=AM(21)                                                     003180
      A(4,1)=AM(4)                                                      003190
      A(4,2)=AM(8)                                                      003200
      A(4,3)=AM(11)                                                     003210
      A(4,4)=AM(13)                                                     003220
      A(4,5)=AM(14)                                                     003230
      A(4,6)=AM(22)                                                     003240
      A(4,7)=AM(23)                                                     003250
      A(5,1)=AM(5)                                                      003260
      A(5,2)=AM(9)                                                      003270
      A(5,3)=AM(12)                                                     003280
      A(5,4)=AM(14)                                                     003290
      A(5,5)=AM(15)                                                     003300
      A(5,6)=AM(24)                                                     003310
      A(5,7)=AM(25)                                                     003320
      A(6,1)=AM(16)                                                     003330
      A(6,2)=AM(18)                                                     003340
      A(6,3)=AM(20)                                                     003350
```

```
      A(6,4)=AM(22)                                                          003360
      A(6,5)=AM(24)                                                          003370
      A(6,6)=AM(26)                                                          003380
      A(6,7)=AM(27)                                                          003390
      A(7,1)=AM(17)                                                          003400
      A(7,2)=AM(19)                                                          003410
      A(7,3)=AM(21)                                                          003420
      A(7,4)=AM(23)                                                          003430
      A(7,5)=AM(25)                                                          003440
      A(7,6)=AM(27)                                                          003450
      A(7,7)=AM(28)                                                          003460
      PRINT 299,((A(J,I),I=1,7),J=1,7)                                       003470
  299 FORMAT(///2X6HA(I,1)4X6HA(I,2)4X6HA(I,3)4X6HA(I,4)4X6HA(I,5)4X6HA(     003480
     1I,6)4X6HA(I,7)///(7E10.4))                                             003490
      DO 117 I=1,7                                                           003500
      BN(I,1)=BM(I)                                                          003510
  117 CN(I,1)=CM(I)                                                          003520
      NN=7                                                                   003530
      MM=1                                                                   003540
      EP=1.E-9                                                               003550
      CALL GAUSS2(NN,MM,EP,A,BN,ZX,KER)                                      003560
      PRINT 799,ZX(1,1),ZX(2,1),ZX(3,1),ZX(4,1),ZX(5,1),ZX(6,1),ZX(7,1)      003570
  799 FORMAT(///11H ZX(1,1) = E14.8//11H ZX(2,1) = E14.8//11H ZX(3,1) =      003580
     1E14.8//11H ZX(4,1) = E14.8//11H ZX(5,1) = E14.8//11H ZX(6,1) = E14     003590
     2.8//11H ZX(7,1) = E14.8//)                                             003600
      CALL GAUSS2(NN,MM,EP,A,CN,XY,KER)                                      003610
      PRINT 759,XY(1,1),XY(2,1),XY(3,1),XY(4,1),XY(5,1),XY(6,1),XY(7,1)      003620
  759 FORMAT(///11H XY(1,1) = E14.8//11H XY(2,1) = E14.8//11H XY(3,1) =      003630
     1E14.8//11H XY(4,1) = E14.8//11H XY(5,1) = E14.8//11H XY(6,1) = E14     003640
     2.8//11H XY(7,1) = E14.8//)                                             003650
C  THIS IS THE END OF PART 6 AND THE START OF PART 7.  PART 7 READS IN       003660
C  THE TITLES THAT ARE PUT ON THE PLOTS DRAWN BY THE PROGRAM.                003670
  456 DO 107 I=1,12                                                          003680
  107 ITITLE(I)=8H                                                           003690
      ITITLE(1)=8H KOCH, J                                                   003700
      ITITLE(2)=8H. W., JR                                                   003710
      ITITLE(7)=8H SIG VS.                                                   003720
      ITITLE(8)=8H PHI                                                       003730
      ITITLE(9)=8HRING NO.                                                   003740
      GO TO (301,302,303,304,305,306,307,308),K                              003750
  301 ITITLE(10)=8H ONE                                                      003760
      GO TO 315                                                              003770
  302 ITITLE(10)=8H TWO                                                      003780
      GO TO 315                                                              003790
  303 ITITLE(10)=8H THREE                                                    003800
      GO TO 315                                                              003810
  304 ITITLE(10)=8H FOUR                                                     003820
      GO TO 315                                                              003830
  305 ITITLE(10)=8H FIVE                                                     003840
      GO TO 315                                                              003850
  306 ITITLE(10)=8H SIX                                                      003860
      GO TO 315                                                              003870
  307 ITITLE(10)=8H SEVEN                                                    003880
      GO TO 315                                                              003890
  308 ITITLE(10)=8H EIGHT                                                    003900
  315 ITITLE(11)=8H 3/7/66                                                   003910
```

```
      LB=4H                                                              003920
      LD=8H                                                              003930
C  THIS IS THE END OF PART 7 AND THE START OF PART 8.  PART 8 COMPUTES    003940
C  THE THEORETICAL VALUES OF NORMAL STRESS AND SHEAR STRESS AROUND THE    003950
C  PIPE DIAMETER AND TRANSFORFS THEM INTO VALUES THAT ARE TO BE PLOTTED.  003960
      ZZZ=6.28318/628.                                                   003970
      DO 26 I=1,628                                                      003980
      ZI=I                                                               003990
      PHI1(I)=ZI*ZZZ                                                     004000
      SO=SINF(PHI1(I))                                                   004010
      CO=COSF(PHI1(I))                                                   004020
      SO2=SINF(2.*PHI1(I))                                               004030
      CO2=COSF(2.*PHI1(I))                                               004040
      SO3=SINF(3.*PHI1(I))                                               004050
      CO3=COSF(3.*PHI1(I))                                               004060
      SIGN7(I)=ZX(1,1)+ZX(2,1)*SO+ZX(3,1)*CO+ZX(4,1)*SO2+ZX(5,1)*CO2+ZX(  004070
     16,1)*SO3+ZX(7,1)*CO3                                               004080
      TAUN7(I)=XY(1,1)+XY(2,1)*SO+XY(3,1)*CO+XY(4,1)*SO2+XY(5,1)*CO2+XY(  004090
     16,1)*SO3+XY(7,1)*CO3                                               004100
      XSNR(I)=100.*CO                                                    004110
      YSNR(I)=100.*SO                                                    004120
      XSN7(I)=(100.+SIGN7(I))*CO                                         004130
      YSN7(I)=(100.+SIGN7(I))*SO                                         004140
      XTNR(I)=40.*CO                                                     004150
      YTNR(I)=40.*SO                                                     004160
      XTN7(I)=(40.+TAUN7(I))*CO                                          004170
      YTN7(I)=(40.+TAUN7(I))*SO                                          004180
   26 CONTINUE                                                           004190
C  THIS IS THE END OF PART 8 AND THE START OF PART 9.  PART 9 CONTAINS    004200
C  THE STATEMENTS WHICH CALL THE SUBROUTINE TO PLOT THE STRESS            004210
C  DISTRIBUTION PATTERN.                                                  004220
      CALL DRAW(628,XSNR,YSNR,1,0,LD,ITITLE,50.,50.,4,4,2,2,8,8,0,LAST)  004230
      CALL DRAW(628,XSN7,YSN7,2,0,LB,ITITLE,50.,50.,4,4,2,2,8,8,0,LAST)  004240
      CALL DRAW(12,XSXX,YSXX,3,1,LD,ITITLE,50.,50.,4,4,2,2,8,8,0,LAST)   004250
      ITITLE(7)=8H TAU VS.                                               004260
      CALL DRAW(628,XTNR,YTNR,1,0,LD,ITITLE,20.,20.,4,4,2,2,8,8,0,LAST)  004270
      CALL DRAW(628,XTN7,YTN7,2,0,LB,ITITLE,20.,20.,4,4,2,2,8,8,0,LAST)  004280
      CALL DRAW(12,XTXY,YTXY,3,1,LD,ITITLE,20.,20.,4,4,2,2,8,8,0,LAST)   004290
C  THIS IS THE END OF PART 9 AND THE START OF PART 10.  PART 10 COMPUTES  004300
C  THE THREE FORCES AND THE THREE MOMENTS AT EACH RING WHICH ARE USED     004310
C  FOR THE EQUILIBRIUM CHECK.                                             004320
  804 XFOR(K)=2.*3.14159*ZX(1,1)*RA(K)*TH(K)                             004330
      YFOR(K)=-3.14159*XY(2,1)*RA(K)*TH(K)                               004340
      ZFOR(K)=3.14159*XY(3,1)*RA(K)*TH(K)                                004350
      XMOM(K)=2.*3.14159*XY(1,1)*RA(K)*RA(K)*TH(K)                       004360
      YMOM(K)=3.14159*ZX(2,1)*RA(K)*RA(K)*TH(K)                          004370
      ZMOM(K)=-3.14159*ZX(3,1)*RA(K)*RA(K)*TH(K)                         004380
   85 CONTINUE                                                           004390
      PRINT 555,(XFOR(K),YFOR(K),ZFOR(K),XMOM(K),YMOM(K),ZMOM(K),K=1,8)  004400
  555 FORMAT(///2X7HXFOR(K)5X7HYFOR(K)5X7HZFOR(K)5X7HXMOM(K)5X7HYMOM(K)5  004410
     1X7HZMOM(K)///(6E12.5))                                             004420
      END                                                                004430
C  THIS IS THE END OF PART 10 AND THE START OF PART 11.  PART 11 IS THE   004440
C  SUBROUTINE GAUSS 2.                                                    004450
```

```
      SUBROUTINE GAUSS2(N,M,EP,A,B,X,KER)                                00010
      DIMENSION A(11,11),B(11,1),X(11,1)                                 00020
      NPM=N+M                                                            00030
      DO 2 K = 1,M                                                       00040
      I = N+K                                                            00050
      DO 2 J = 1,N                                                       00060
    2 A(J,I) = B(J,K)                                                    00070
   10 DO 34 L=1,N                                                        00080
      KP=0                                                               00090
      Z=0.0                                                              00100
      DO 12 K=L,N                                                        00110
      IF(Z-ABSF(A(K,L)))11,12,12                                         00120
   11 Z=ABSF(A(K,L))                                                     00130
      KP=K                                                               00140
   12 CONTINUE                                                           00150
      IF(L-KP)13,20,20                                                   00160
   13 DO 14 J=L,NPM                                                      00170
      Z=A(L,J)                                                           00180
      A(L,J)=A(KP,J)                                                     00190
   14 A(KP,J)=Z                                                          00200
   20 IF(ABSF(A(L,L))-EP)50,50,30                                        00210
   30 IF(L-N)31,40,40                                                    00220
   31 LP1=L+1                                                            00230
      DO 34 K=LP1,N                                                      00240
      IF(A(K,L))32,34,32                                                 00250
   32 RATIO=A(K,L)/A(L,L)                                                00260
      DO 33 J=LP1,NPM                                                    00270
   33 A(K,J)=A(K,J)-RATIO*A(L,J)                                         00280
   34 CONTINUE                                                           00290
   40 DO 43 I=1,N                                                        00300
      II=N+1-I                                                           00310
      DO 43 J=1,M                                                        00320
      JPN=J+N                                                            00330
      S=0.0                                                              00340
      IF(II-N)41,43,43                                                   00350
   41 IIP1=II+1                                                          00360
      DO 42 K=IIP1,N                                                     00370
   42 S=S+A(II,K)*X(K,J)                                                 00380
   43 X(II,J)=(A(II,JPN)-S)/A(II,II)                                     00390
      KER=1                                                              00400
      RETURN                                                             00410
   50 KER = 2                                                            00420
      END                                                                00430
C  THIS IS THE END OF PART 11 AND THE START OF PART 12.  PART 12 IS THE   004460
C  SUBROUTINE DRAW.                                                       004470
      SUBROUTINE DRAW (NUMPTS, X, Y, MODCURV, ITYPE, LABEL,              00000000
     1                 ITITLE, EXSCALE, YSCALE, IXUP, IYRIGHT,           00000010
     2                  MODEXAX, MODEYAX, IWIDE, IHIGH, IGRID,           00000020
     3                 LAST)                                                  30
C                                                                          0    40
C                                                                          0    50
C     A GENERAL CURVE DRAWING AND POINT PLOTTING SUBROUTINE                0000006
C          PROGRAMMER   J. R. WARD                                         00000070
C          DATE         FEB. 1964, REVISED JUNE 1965                       00000080
C          SYSTEM       FORTRAN 60                                         00000090
C          OUTPUT       LOGICAL TAPE NUMBER 8                              00000100
```

```
C       NOTE          ASTERISKS MARK CHANGES FOR FORTRAN 63                 00000011
C                                                                           000* 120
C   INPUT ARGUMENTS ---                                                     00000013
C                                                                           0000 140
C       1. NUMPTS     NUMBER OF POINTS TO BE PLOTTED. THIS MUST ALWAYS      00000150
C                     BE AT LEAST 2, AND MUST NOT EXCEED 30 FOR POINT       00000160
C                     PLOTTING, OR 900 FOR CURVE DRAWING.                   00000170
C                                                                           000* 180
C       2. X          ARRAY OF X-ORDINATES. DIMENSION AT LEAST EQUAL        00000190
C                     TO NUMPTS AND NOT MORE THAN 900 IN CALLING            00000200
C                     PROGRAM.                                              00000210
C                                                                           000* 220
C       3. Y          ARRAY OF Y-ORDINATES. DIMENSION AS FOR THE            00000230
C                     X ARRAY IN THE CALLING PROGRAM.                       00000240
C                                                                           000  250
C       4. MODCURV    CONTROLS THE NUMBER OF CURVES, AND/OR SETS OF         00000260
C                     POINTS, ON EACH GRAPH. THE CODES ARE                  00000270
C                           0    ONLY ONE PLOT ON THIS GRAPH                00000280
C                           1    FIRST PLOT ON MULTI-PLOT GRAPH             00000290
C                           2    INTERMEDIATE PLOT ON MULTI-PLOT GRAPH      00000300
C                           3    LAST PLOT ON MULTI-PLOT GRAPH.             00000310
C                                                                           000  320
C       5. ITYPE      CONTROLS THE TYPE OF PLOT. THE CODES ARE              00000330
C                           0    STRAIGHT LINES JOIN SUCCESSIVE POINTS      00000340
C                                (STANDARD CURVE DRAWING)                   00000350
C                           1    POINTS PLOTTED WITH CROSS (X)              00000360
C                           2    POINTS PLOTTED WITH PLUS (+)               00000370
C                           3    POINTS PLOTTED WITH SQUARE                 00000380
C                           4    POINTS PLOTTED WITH DIAMOND                00000390
C                           5    POINTS PLOTTED WITH TRIANGLE               00000400
C                     WHEN POINTS ARE BEING PLOTTED (ITYPE=1 THRU. 5),      00000410
C                     THE POINTS ARE NOT CONNECTED.                         00000420
C                                                                           000  430
C       6. LABEL      IF A CURVE IS BEING DRAWN (ITYPE = 0), LABEL IS       00000440
C                     A 4-CHARACTER BCD CURVE IDENTIFIER WHICH WILL BE      00000450
C                     REPRODUCED AT THE END OF THE CURVE. FOR EXAMPLE,      00000460
C                     LABEL = 4H ONE . IF POINTS ARE BEING PLOTTED,         00000470
C                     LABEL IS AN 8-CHARACTER IDENTIFIER. THE FIRST 4       00000480
C                     CHARACTERS ARE REPRODUCED WITH THE FIRST PLOTTED      00000490
C                     POINT, AND THE LAST 4 WITH THE LAST POINT. SET        00000500
C                     TO BLANK ANY UNWANTED CHARACTERS.                     00000510
C                                                                           000- 520
C       7. ITITLE     AN ARRAY OF TWELVE 8-CHARACTER BCD WORDS.             00000530
C                     THE FIRST SIX WORDS WILL BE REPRODUCED AS THE         00000540
C                     FIRST LINE OF GRAPH TITLE, AND THE LAST SIX           00000550
C                     WORDS WILL FORM THE SECOND LINE. THE TITLE MUST       00000560
C                     INCLUDE THE USERS JOB IDENTIFICATION. DIMENSION       00000570
C                     12 IN CALLING PROGRAM, AND SET TO BLANK ALL           00000580
C                     UNWANTED CHARACTERS.                                  00000590
C                                                                           000  600
C       8. EXSCALE    X-SCALE (UNITS/INCH) AS POSITIVE FLOATING POINT       00000610
C                     VARIABLE WITH ONE FIGURE SIGNIFICANCE. SET TO         00000620
C                     ZERO FOR AUTO-SCALE.                                  00000630
C                                                                           000  640
C       9. YSCALE     Y-SCALE (UNITS/INCH) AS POSITIVE FLOATING POINT       00000650
C                     VARIABLE WITH ONE FIGURE SIGNIFICANCE. SET TO         00000660
```

```
C                       ZERO FOR AUTO-SCALE.                                    00000670
C                                                                               000  680
C       10. IXUP        X-AXIS OFFSET FROM BOTTOM OF GRAPH IN INCHES.           00000690
C                       THIS MUST NOT EXCEED IHIGH, AND MUST NOT BE             00000700
C                       NEGATIVE.                                               00000710
C                                                                               000* 720
C       11. IYRIGHT     Y-AXIS OFFSET FROM LEFT OF GRAPH IN INCHES. THIS        00000730
C                       MUST NOT EXCEED IWIDE, AND MUST NOT BE NEGATIVE.        00000740
C                                                                               000  750
C       12. MODEXAX     MODE OF X-AXIS OFFSET. SEE CODES BELOW.                 00000760
C                                                                               000  770
C       13. MODEYAX     MODE OF Y-AXIS OFFSET. THE CODES ARE AS FOLLOWS         00000780
C                             0    COMPUTED OFFSET, HOLDING ORIGIN ON           00000790
C                                  GRAPH. THE CORRESPONDING IXUP OR             00000800
C                                  IYRIGHT IS IGNORED                           00000810
C                             1    COMPUTED OFFSET, WITH ORIGIN OFF THE         00000820
C                                  GRAPH IF APPROPRIATE. THE CORRESPOND-        00000830
C                                  ING IXUP OR IYRIGHT IS IGNORED. USE          00000840
C                                  ONLY WITH AUTO-SCALE                         00000850
C                             2    AXIS OFFSET AS SPECIFIED BY IXUP OR          00000860
C                                  IYRIGHT.                                     00000870
C                                                                               000* 880
C       14. IWIDE       WIDTH OF GRAPH IN INCHES. THIS MUST NOT EXCEED          00000890
C                       NINE.  ZERO WILL BE READ AS EIGHT INCHES.               00000900
C                                                                               000  910
C       15. IHIGH       HEIGHT OF GRAPH IN INCHES. THIS MUST NOT EXCEED         00000920
C                       FIFTEEN. ZERO WILL BE READ AS EIGHT INCHES.             00000930
C                                                                               000  940
C       16. IGRID       IF SET TO 1, A ONE INCH BY ONE INCH GRID WILL           00000950
C                       BE SUPERIMPOSED ON THE GRAPH.                           00000960
C                                                                               0000 970
C       17. LAST        INDICATES TO CALLING PROGRAM WHETHER LAST PLOT          00000980
C                       WAS COMPLETED SUCCESSFULLY. THE CODES ARE               00000990
C                             0    LAST PLOT COMPLETED SUCCESSFULLY             00001000
C                             1    LAST PLOT NOT SUCCESSFUL                     00001010
C                             2    LAST PLOT NOT SUCCESSFUL AND NO              00001020
C                                  FURTHER GRAPH OUTPUT WILL BE ATTEMPT-        00001030
C                                  ED UNTIL MODCURV IS NEXT ONE OR ZERO         00001040
C                             3    DRAW WAS ENTERED WITH MODCURV NOT            00001050
C                                  EQUAL TO ONE OR ZERO WHILE THE ERROR         00001060
C                                  LOCK-OUT WAS SET.                            00001070
C                       THIS ARGUMENT MUST ALWAYS BE A NAME IN THE CALL         00001080
C                       STATEMENT. NEVER A NUMBER.                              00001090
C                                                                               00001100
C                                                                               00001110
C     NOTE ---                                                                  00001120
C         ALL ARGUMENTS FROM NUMBER 7 THRU. NUMBER 16 ARE IGNORED WHEN          00001130
C         MODCURV IS EITHER 2 OR 3. HOWEVER, ARGUMENTS MUST NEVER BE            00001140
C         OMITTED FROM THE CALLING STATEMENT. IT IS MERELY THEIR VALUES         00001150
C         WHICH ARE THEN IRRELEVANT. ARGUMENTS MAY BE LISTED BY NAME OR         00001160
C         VALUE IN THE CALL STATEMENT. NO VALUE IN THE CALLING PROGRAM          00001170
C         WILL BE ALTERED BY THIS SUBROUTINE.                                   00001180
C                                                                               00001190
C     REFERENCE ---                                                             00001200
C         THE BINARY TAPE FORMAT REQUIRED BY THE OFF-LINE PLOTTER IS            00001210
C         DESCRIBED IN THE WRITEUP OF THE CDC 160A GRAPH PLOT PROGRAM           00001220
```

```
C          (IDENT, 9001). THE FORMAT REQUIRED BY THE CDC 160 PROGRAM IS   00001230
C          SIMILAR EXCEPT THAT THE INTERPOLATION ARGUMENT MUST BE ZERO.   00001240
C                                                                         00001250
C                                                                         00001260
      DIMENSION X(900), Y(900), ITITLE(12), JXTIT(12), JYTIT(12),         00001270
     1          LTITLE(14), KAXIS(5), ICURV(460), JGRID(25), ICONT(1),    00001280
     2          JJTITLE(12)                                              00001290
      IPOINT = ITYPE                                                     00001300
      CON(           ICONTRL = 40000B).                                  00001310
      CON(ICURV3 = 2777377720202020B, ICURV4 = 0104000000000003).        00001320
*        REPLACE WITH DATA STATEMENT IN FORTRAN 62-3.                     00001330
*       PUT ITEST = 0 IN DATA STATEMENT.                                  00001340
      IF (JTEST - 73546912)  9070,9071,9070                              00001350
 9070 ITEST = 0                                                          00001360
      JTEST = 73546912                                                   00001370
 9071 CONTINUE                                                           00001380
*        REMOVE ABOVE NONSENSE IN FORTRAN 63.                             00001390
C         CHECK PREVIOUS OPERATION OF ROUTINE, IF ANY. CODES ARE          00001400
C             ITEST = 0     IF PREVIOUS GRAPH, IF ANY, COMPLETED        00001410
C             ITEST = 1     IF PREVIOUS GRAPH NOT COMPLETED             00001420
C             ITEST = 2     IF ERROR FOUND WHILE MODCURV WAS ONE, OR IF 00001430
C                           MODCURV WAS ILLEGAL.                        00001440
      IF(ITEST - 2)1000,1001,1000                                        00001450
 1001 IF(MODCURV)1003,1002,1003                                          00001460
 1002 ITEST = 0                                                          00001470
      GO TO 1000                                                         00001480
 1003 IF(MODCURV - 1)1004,1002,1004                                      00001490
 1004 LAST = 3                                                           00001500
      RETURN                                                             00001510
C         SET UP ERROR RETURN ROUTINE. ENTRY AT STATEMENT 1005.          00001520
 1005 IF(ITEST)1009,1006,1009                                            00001530
 1006 IF(MODCURV)1007,1008,1007                                          00001540
 1007 PRINT 1100                                                         00001550
 1100 FORMAT ( 59H NO FURTHER GRAPH OUTPUT UNTIL MODCURV NEXT IS ZERO OR00001560
     1 ONE. ,/)                                                          00001570
      ITEST = 2                                                          00001580
      LAST = 2                                                           00001590
      RETURN                                                             00001600
 1008 PRINT 1101                                                         00001610
 1101 FORMAT ( 30H THIS PLOT WILL NOT BE OUTPUT. ,/)                     00001620
      LAST = 1                                                           00001630
      RETURN                                                             00001640
 1009 IF(MODCURV - 2)1010,1008,1010                                      00001650
 1010 IF(MODCURV - 3)1007,1011,1007                                      00001660
 1011 ITEST = 0                                                          00001670
      GO TO 1008                                                         00001680
C         CHECK LEGALITY OF INPUT ARGUMENTS.                             00001690
 1000 IF(NUMPTS - 2)1,2,2                                                00001700
    1 PRINT 100                                                          00001710
  100 FORMAT (/, 32H NUMPTS MUST NOT BE LESS THAN 2. )                   00001720
      GO TO 1005                                                         00001730
    2 IF(IPOINT)9000,9004,9001                                           00001740
 9000 PRINT 9100                                                         00001750
 9100 FORMAT (/, 15H ILLEGAL ITYPE. )                                    00001760
      GO TO 1005                                                         00001770
 9001 IF(IPOINT - 5)9002,9002,9000                                       00001780
```

```
9002 IF(NUMPTS - 30)3,3,9003                                                    00001790
9003 PRINT 9101                                                                 00001800
9101 FORMAT (/, 46H NUMPTS MUST NOT EXCEED 30 FOR POINT PLOTTING. )             00001810
     GO TO 1005                                                                 00001820
9004 IF(NUMPTS - 900)3,3,9005                                                   00001830
9005 PRINT 9102                                                                 00001840
9102 FORMAT (/, 28H NUMPTS MUST NOT EXCEED 900. )                               00001850
     GO TO 1005                                                                 00001860
   3 IX = 1HX                                                                   00001870
     IY = 1HY                                                                   00001880
     AMAXX = -0.2E+100                                                          00001890
     AMAXY = -0.2E+100                                                          00001900
     AMINX = +0.2E+100                                                          00001910
     AMINY = +0.2E+100                                                          00001920
     DO 1020 I= 1,NUMPTS                                                        00001930
     AMAXX = MAX1F(X(I),AMAXX)                                                  00001940
     AMAXY = MAX1F(Y(I),AMAXY)                                                  00001950
     AMINX = MIN1F(X(I),AMINX)                                                  00001960
1020 AMINY = MIN1F(Y(I),AMINY)                                                  00001970
     AMAXA = MAX1F(ABSF(AMAXX), ABSF(AMAXY), ABSF(AMINX), ABSF(AMINY))          00001980
     IF(AMAXA - 1.0E+99)1022,1022,1021                                          00001990
1021 PRINT 1102                                                                 00002000
1102 FORMAT (/, 58H NO X OR Y VALUE MAY EXCEED 1.0E+99 IN ABSOLUTE MAGN00002010
    1ITUDE. )                                                                   00002020
     GO TO 1005                                                                 00002030
1022 IF(ABSF(AMAXX - AMINX) - 1.0E-97)1023,1025,1025                            00002040
1023 IF(ABSF(AMAXY - AMINY) - 1.0E-97)1024,1025,1025                            00002050
1024 PRINT 1103                                                                 00002060
1103 FORMAT (/, 38H ALL POINTS HAVE THE SAME COORDINATES. )                     00002070
     GO TO 1005                                                                 00002080
1025 IF(ITEST)4,7,4                                                             00002090
   4 IF(MODCURV - 2)5,240,5                                                     00002100
   5 IF(MODCURV - 3)6,240,6                                                     00002110
   6 PRINT 101                                                                  00002120
 101 FORMAT (/, 17H ILLEGAL MODCURV. )                                          00002130
     GO TO 1005                                                                 00002140
   7 IF(MODCURV)6,9,8                                                           00002150
   8 IF(MODCURV - 1)6,9,6                                                       00002160
   9 IF(IWIDE)10,11,12                                                          00002170
  10 ITIT = 5HIWIDE                                                             00002180
     PRINT 102, ITIT, ITIT                                                      00002190
 102 FORMAT (/,  9H ILLEGAL ,A5,29H. GRAPH WILL BE PLOTTED WITH ,A5,            00002200
    1            5H = 8. ,/)                                                  00002210
  11 JWIDE = 8                                                                  00002220
     GO TO 14                                                                   00002230
  12 IF(IWIDE - 9)13,13,10                                                      00002240
  13 JWIDE = IWIDE                                                              00002250
  14 IF(IHIGH)15,16,17                                                          00002260
  15 ITIT = 5HIHIGH                                                             00002270
     PRINT 102, ITIT, ITIT                                                      00002280
  16 JHIGH = 8                                                                  00002290
     GO TO 19                                                                   00002300
  17 IF(IHIGH - 15)18,18,15                                                     00002310
  18 JHIGH = IHIGH                                                              00002320
  19 NODEXAX = MODEXAX                                                          00002330
     IF(MODEXAX)20,27,21                                                        00002340
```

```
   20 ITIT= 8HMODEXAX.                                                    00002350
      PRINT 104, ITIT, IX                                                 00002360
  104 FORMAT (/,  9H ILLEGAL ,A8, 32H GRAPH WILL BE PLOTTED WITH MODE,    00002370
     1        A1,  7HAX = 0, ,/)                                          00002380
      NODEXAX = 0                                                         00002390
      GO TO 27                                                            00002400
   21 IF(MODEXAX - 2)27,22,22                                             00002410
   22 IF(IXUP - JHIGH)24,24,23                                            00002420
   23 ITIT = 8HIXUP.                                                      00002430
      PRINT 104, ITIT, IX                                                 00002440
      NODEXAX = 0                                                         00002450
      GO TO 27                                                            00002460
   24 IF(IXUP)23,26,26                                                    00002470
   26 JXUP = IXUP                                                         00002480
   27 NODEYAX = MODEYAX                                                   00002490
      IF(MODEYAX)28,35,29                                                 00002500
   28 ITIT=8HMODEYAX.                                                     00002510
      PRINT 104, ITIT, IY                                                 00002520
      NODEYAX = 0                                                         00002530
      GO TO 35                                                            00002540
   29 IF(MODEYAX - 2)35,30,28                                             00002550
   30 IF(IYRIGHT - JWIDE)32,32,31                                         00002560
   31 ITIT = 8HIYRIGHT.                                                   00002570
      PRINT 104, ITIT, IY                                                 00002580
      NODEYAX = 0                                                         00002590
      GO TO 35                                                            00002600
   32 IF(IYRIGHT) 31,34,34                                                00002610
   34 JYRIGHT = IYRIGHT                                                   00002620
C          INITIALIZE PRIOR TO SCALING AND AXIS LOCATING.                 00002630
C          IFLAG = 0 FOR PASS WITH XDATA. IFLAG = 1 FOR PASS WITH YDATA.  00002640
   35 DO 2235 IOTA=1,12                                                   00002650
 2235 JJTITLE(IOTA) = ITITLE(IOTA)                                        00002660
      IFLAG = 0                                                           00002670
      BETA = 0.                                                           00002680
      SCALE = EXSCALE                                                     00002690
      IAXIS = JYRIGHT                                                     00002700
      MODE = NODEYAX                                                      00002710
      ISIZE = JWIDE                                                       00002720
      IXY = IX                                                            00002730
      IYX = IY                                                            00002740
      AMAX = AMAXX                                                        00002750
      AMIN = AMINX                                                        00002760
      GO TO 52                                                            00002770
   50 IFLAG = 1                                                           00002780
      BETA = 0.                                                           00002790
      SCALE = YSCALE                                                      00002800
      IAXIS = JXUP                                                        00002810
      MODE  = NODEXAX                                                     00002820
      ISIZE = JHIGH                                                       00002830
      AMAX = AMAXY                                                        00002840
      AMIN = AMINY                                                        00002850
      IXY = IY                                                            00002860
      IYX  = IX                                                           00002870
C          CHECK SCALE AND GO TO FIXED OR AUTO SCALE ROUTINES.            00002880
   52 IF(SCALE)53,59,56                                                   00002890
   53 PRINT 114, IXY, IXY                                                 00002900
```

```
  114 FORMAT (/,  2H ILLEGAL ,A1,24HSCALE. GRAPH WILL BE PLOTTED WITH A 00002310
     1TO ,A1, 7H-SCALE. ,/)                                             00002320
      GO TO 59                                                          00002330
C     EXPRESS FIXED SCALE IN E FORMAT WITH ONE FIGURE SIGNIFICANCE.     00002340
   56 IF(SCALE - 1.0E+99)57,57,58                                       00002350
   57 IF(SCALE - 1.0E-99)58,58,59                                       00002360
   58 CALL SCALEIT(SCALE,ISCALE ,FACTOR,1)                              00002370
      SCALE = FACTOR*10.**ISCALE                                        00002380
C         CHECK AND COMPUTE AXIS LOCATION IF NECESSARY. FIXED SCALE     00002390
C         CASE. ITAG = 0 IF ORIGIN ON GRAPH OR 1 IF IT IS SUPPRESSED.   00002400
      IF(MODE - 1)1032,1031,1032                                        00002410
 1030 ITAG =0                                                           00002420
      GO TO 203                                                         00002430
 1031 PRINT 1104 , IYX, IXY, IXY                                        00002440
 1104 FORMAT (/, 5H MODE,A1,24HAX MUST NOT BE 1 UNLESS ,A1,57HSCALE IS  00002450
     1 (AUTO-SCALE). GRAPH WILL BE PLOTTED WITH AUTO ,A1, 7H-SCALE. ,/) 00002460
      GO TO 59                                                          00002470
 1032 IF(ABSF(AMAX - AMIN) - 1.0E-97)1033,1038,1038                     00002480
 1033 IF(ABSF(AMAX) - 1.0E-97)1034,1039,1039                            00002490
 1039 IF(AMAX)1036,1034,1037                                            00002500
 1034 IAXIS = ISIZE/2                                                   00002510
      GO TO 1030                                                        00002520
 1036 IAXIS = ISIZE                                                     00002530
      GO TO 1030                                                        00002540
 1037 IAXIS = 0                                                         00002550
      GO TO 1030                                                        00002560
 1038 IF(SIGNF(1.,AMAX) - SIGNF(1.,AMIN))1040,1039,1040                 00002570
 1040 ASIZE = ISIZE                                                     00002580
      IAXIS = -AMIN/(AMAX - AMIN)*ASIZE +0.5                            00002590
      GO TO 1030                                                        00002600
C         AUTO SCALE ROUTINE.                                           00002610
   59 IF(MODE - 1)60,64,59                                              00002620
   60 AMAX = MAX1F(0., AMAX)                                            00002630
      AMIN = MIN1F(0., AMIN)                                            00002640
   64 IF(ABSF(AMAX - AMIN) - 1.0E-97)65,68,68                           00002650
   65 PRINT 116, IXY, IXY, IYX                                          00002660
  116 FORMAT (/,  5H ALL ,A1,47H VALUES EQUAL. AUTO SCALE POSSIBLE ONLY 00002670
     1IF THE ,A1,29H VALUES ARE NON-ZERO AND MODE,A1, 7HAX = 2. )       00002680
      GO TO 1005                                                        00002690
   68 ASIZE = ISIZE                                                     00002700
      SCALE = (AMAX - AMIN)/ASIZE                                       00002710
      GO TO 83                                                          00002720
   69 IF(ABSF(AMAX - AMIN) - 1.0E-97)70,74,74                           00002730
   70 IF(ABSF(AMAX) - 1.0E-97)71,74,74                                  00002740
   71 PRINT 118, IXY                                                    00002750
  118 FORMAT (/,  5H ALL ,A1,38H VALUES ZERO. AUTO SCALE NOT POSSIBLE. )00002760
      GO TO 1005                                                        00002770
   74 IF(AMAX - 1.0E-97) 76,75,75                                       00002780
   75 IF(ISIZE - IAXIS)77,76,77                                         00002790
   76 SCALE1 = 0.                                                       00002800
      GO TO 78                                                          00002810
   77 AXIS = IAXIS                                                      00002820
      ASIZE = ISIZE                                                     00002830
      SCALE1 = AMAX/(ASIZE - AXIS)                                      00002840
   78 IF(AMIN + 1.0E-97)79,79,80                                        00002850
   79 IF(IAXIS)81,80,81                                                 00002860
```

```
   80 SCALE2 = 0.                                                        00003470
      GO TO 82                                                           00003480
   81 AXIS = IAXIS                                                       00003490
      SCALE2 = -AMIN/AXIS                                                00003500
   82 IF(SCALE1 + SCALE2)1984,1982,1984                                  00003510
 1982 PRINT 1983, IYX, IYX                                               00003520
 1983 FORMAT (/, 56H NONE OF THE PLOT LIES ON THE GRAPH WITH THIS SPECIFC00003530
     1IED ,A1,47H-AXIS LOCATION. GRAPH WILL BE PLOTTED WITH MODE,A1,     00003540
     2  7HAX = 0.  ,/)                                                  00003550
      MODE = 0                                                           00003560
      GO TO 60                                                           00003570
 1984 SCALE = MAX1F(SCALE1, SCALE2)                                      00003580
   83 CALL SCALFIT(SCALE, ISCAL10,FACTOR,3)                              00003590
      IF(FACTOR - 5.05)85,85,84                                          00003600
   84 FACTOR = 1                                                         00003610
      ISCAL10 = ISCAL10 + 1                                              00003620
      GO TO 90                                                           00003630
   85 IF(FACTOR - 2.02)87,87,86                                          00003640
   86 FACTOR = 5                                                         00003650
      GO TO 90                                                           00003660
   87 IF(FACTOR - 1.01)89,89,88                                          00003670
   88 FACTOR = 2                                                         00003680
      GO TO 90                                                           00003690
   89 FACTOR = 1                                                         00003700
   90 SCALE = FACTOR*10.**ISCAL10                                        00003710
C          COMPUTE AXIS LOCATION IF NECESSARY. AUTO SCALE CASE.         00003720
      IF(MODE - 1)92,91,93                                               00003730
   91 IF(SIGNF(1.,AMAX) - SIGNF(1.,AMIN))92,94,92                        00003740
   92 IAXIS = -AMIN/SCALE + 0.5                                          00003750
   93 ITAG = 0                                                           00003760
      GO TO 203                                                          00003770
   94 IF(AMAX)95,95,200                                                  00003780
   95 IAXIS = ISIZE                                                      00003790
      BETA = -AMAX/SCALE                                                 00003800
      IF(BETA - 1.E+12)99,99,96                                          00003810
   96 PRINT 120, IXY                                                     00003820
  120 FORMAT (/, 15H THE ORIGIN OF ,A1, 43H CANNOT BE OFFSET MORE THAN 100003830
     1.0E+12 INCHES. )                                                   00003840
      GO TO 1005                                                         00003850
   99 IBETA = BETA + 0.5                                                 00003860
      BETA = -IBETA                                                      00003870
C          BETA IS THE NUMBER OF INCHES OF ORIGIN SUPPRESSION,POSITIVE IF00003880
C          TRUE ORIGIN IS BELOW OR TO LEFT OF THE GRAPH.                00003890
      IF(BETA + 1.)97,97,93                                              00003900
   97 ITAG = 1                                                           00003910
      GO TO 203                                                          00003920
  200 IAXIS = 0                                                          00003930
      BETA = AMIN/SCALE                                                  00003940
      IF(BETA - 1.E+12)201,201,96                                        00003950
  201 IBETA = BETA + 0.5                                                 00003960
      BETA = IBETA                                                       00003970
      IF(BETA - 1.)93,202,202                                            00003980
  202 ITAG = 1                                                           00003990
C          RELEASE RESULTS TO REMAINING PART OF PROGRAM. START SECOND   00004000
C          PASS FOR Y VALUES IF NOT YET COMPUTED.                       00004010
  203 IF(IFLAG)205,204,205                                               00004020
```

```
  204 SCALEX = SCALE                                                    00004030
      IXFACT  = FACTOR                                                  00004040
      IXSCIO = ISCALIO                                                  00004050
      IXAXIS = IAXIS                                                    00004060
      ITAGX  = ITAG                                                     00004070
      ISIZEX = ISIZE                                                    00004080
      BETAX = BETA                                                      00004090
      GO TO 50                                                          00004100
  205 BETAY = BETA                                                      00004110
      SCALEY = SCALE                                                    00004120
      IYFACT =  FACTOR                                                  00004130
      IYSCIO = ISCALIO                                                  00004140
      IYAXIS = IAXIS                                                    00004150
C         NOW WRITE RECORDS.                                            00004160
      ITAGY  = ITAG                                                     00004170
      ISIZEY = ISIZE                                                    00004180
C         THIS COMPLETES CALCULATION OF SCALE FACTORS ETC. NOW GENERATE 00004190
C         TAPE RECORDS. FIRST, THE SCALE FACTOR TITLES.                 00004200
  206 JXTIT(1) = 8H1      X-                                            00004210
      JXTIT(2) = 8HSCALE =                                              00004220
      JXTIT(3) = ICODE(SCALEX)                                          00004230
      JXTIT(4) = 8H UNITS/I                                             00004240
      JXTIT(5) = 8HNCH.                                                 00004250
      JYTIT(1) = 8H1      Y-                                            00004260
      JYTIT(2) = 8HSCALE =                                              00004270
      JYTIT(3) = ICODE(SCALEY)                                          00004280
      JYTIT(4) = 8H UNITS/I                                             00004290
      JYTIT(5) = 8HNCH.                                                 00004300
      DO 9206 I=6,11                                                    00004310
      JXTIT(I) = 8H                                                     00004320
 9206 JYTIT(I) = 8H                                                     00004330
      IF(ITAGX)211,211,207                                              00004340
  207 IF(BETAX)208,208,209                                              00004350
  208 JXTIT(7) = 8H   ADD -                                             00004360
      GO TO 210                                                         00004370
  209 JXTIT(7) = 8H   ADD +                                             00004380
  210 JXTIT(8) = ICODE (BETAX*SCALEX)                                   00004390
      JXTIT(9) = 8H UNITS T                                             00004400
      JXTIT(10)= 8HO ALL X                                              00004410
      JXTIT(11)= 8HVALUES.                                              00004420
  211 IF(ITAGY)216,216,212                                              00004430
  212 IF(BETAY)213,213,214                                              00004440
  213 JYTIT(7) = 8H   ADD -                                             00004450
      GO TO 215                                                         00004460
  214 JYTIT(7) = 8H   ADD +                                             00004470
  215 JYTIT(8) = ICODE (BETAY*SCALEY)                                   00004480
      JYTIT(9) = 8H UNITS T                                             00004490
      JYTIT(10)=8HO ALL Y                                               00004500
      JYTIT(11)= 8HVALUES.                                              00004510
  216 ICONT(1) = ICONTRL + 4                                            00004520
C         INSERT TITLE SIZE (02B) AHEAD OF MAIN TITLE RECORD.           00004530
      CALL ISHIFT6 (ITITLE, LTITLE)                                     00004540
C        TEST FOR ALL BLANK TITLES.                                     00004550
      ICHECK = 8H                                                       00004560
      DO 9075 I=1,6                                                     00004570
      IF(ITITLE(I)-ICHECK) 9074,9075,9074                               00004580
```

```
 9074 IF(ITITLE(I) ) 9080,9075,9080                                    00004590
 9075 CONTINUE                                                         00004600
      IT1 = 1                                                          00004610
      ICONT(1) = ICONT(1) - 1                                          00004620
      GO TO 9081                                                       00004630
 9080 IT1 = 0                                                          00004640
 9081 DO 9085 I=7,12                                                   00004650
      IF (ITITLE(I) - ICHECK) 9084,9085,9084                           00004660
 9084 IF (ITITLE(I))9090,9085,9090                                     00004670
 9085 CONTINUE                                                         00004680
      IT2 = 1                                                          00004690
      ICONT(1) = ICONT(1) - 1                                          00004700
      GO TO 9091                                                       00004710
 9090 IT2 = 0                                                          00004720
C          NOW GENERATE AXES RECORDS.                                  00004730
 9091 LFTMGN = 0*100                                                   00004740
      IBOTMGN = 1*100                                                  00004750
      IH = LFTMGN                                                      00004760
      JH = IBOTMGN +IYAXIS*100                                         00004770
      LH = ISIZEX*100                                                  00004780
      IHL = LFTMGN + ISIZEX*100 - 107                                  00004790
      KAXIS(1) = IPACK12(IH,JH,LH,IHL)                                 00004800
      JHL = JH - 13                                                    00004810
      IHL2 = -100                                                      00004820
      IVH  = (ISIZEX -IXAXIS - 1)*IXFACT                               00004830
      IVH2 = -IXFACT                                                   00004840
      KAXIS(2) = IPACK12(JHL,IHL2,IVH,IVH2)                            00004850
      NH = ISIZEX                                                      00004860
      ISH = 8H       14                                                00004870
      IV = LFTMGN + IXAXIS*100                                         00004880
      JV = IBOTMGN                                                     00004890
      KAXIS(3) = IPACK12(NH,ISH,IV,JV)                                 00004900
      LV = ISIZEY*100                                                  00004910
      IVL = IV - 3                                                     00004920
      JVL = IBOTMGN + ISIZEY*100 - 107                                 00004930
      JVL2 = -100                                                      00004940
      KAXIS(4) = IPACK12(LV,IVL,JVL,JVL2)                              00004950
      IVV = (ISIZEY - IYAXIS - 1)*IYFACT                               00004960
      IVV2 = -IYFACT                                                   00004970
      INV = ISIZEY                                                     00004980
      ISV = 8H       11                                                00004990
      KAXIS(5) = IPACK12(IVV,IVV2,INV,ISV)                             00005000
C          NOW GENERATE CURVES.                                        00005010
      SCX = 100./SCALEX                                                00005020
      SCY = 100./SCALEY                                                00005030
      EXAXIS= IXAXIS*100                                               00005040
      YAXIS = IYAXIS*100                                               00005050
      ALFTMGN = LFTMGN                                                 00005060
      BOTMGN = IBOTMGN                                                 00005070
      SHIFTX =EXAXIS - BETAX*100. + ALFTMGN                            00005080
      SHIFTY = YAXIS - BETAY*100. + BOTMGN                             00005090
      EXSIZE= ISIZEX*100 + LFTMGN + 60                                 00005100
      SIZEX = LFTMGN - 60                                              00005110
      YSIZE = ISIZEY*100 + IBOTMGN + 70                                00005120
      SIZEY = IBOTMGN - 70                                             00005130
      ICURV(1) = 0                                                     00005140
```

```
  240 IF(IPOINT)9010,9007,9010                                          00005150
 9007 IF(XMODF(NUMPTS,2))9700,9701,9700                                 00005160
 9700 ISWITCH = 1                                                       00005170
      GO TO 242                                                         00005180
 9701 ISWITCH = 2                                                       00005190
  242 INUM = (NUMPTS + 1)/2                                             00005200
      DO 244 I=1,INUM                                                   00005210
      C1 = X(2*I-1)*SCX + SHIFTX                                        00005220
      C2 = Y(2*I-1)*SCY + SHIFTY                                        00005230
      IF(I-INUM)241,9241,241                                            00005240
 9241 GO TO (9242,241),ISWITCH                                          00005250
 9242 C3 = C1                                                           00005260
      C4 = C2                                                           00005270
      GO TO 9243                                                        00005280
  241 C3 = X(2*I)*SCX + SHIFTX                                          00005290
      C4 = Y(2*I)*SCY + SHIFTY                                          00005300
 9243 C1 = MIN1F(C1,EXSIZE)                                             00005310
      IC1= MAX1F(C1, SIZEX)                                             00005320
      C2 = MIN1F(C2, YSIZE)                                             00005330
      IC2= MAX1F(C2, SIZEY)                                             00005340
      C3 = MIN1F(C3,EXSIZE)                                             00005350
      IC3= MAX1F(C3, SIZEX)                                             00005360
      C4 = MIN1F(C4, YSIZE)                                             00005370
      IC4= MAX1F(C4, SIZEY)                                             00005380
  244 ICURV(I+1) = IPACK12(IC1,IC2,IC3,IC4)                             00005390
      II = INUM + 3                                                     00005400
  246 CALL IPACKL1(LABEL, LABEL1, IDUMMY)                               00005410
      ICURV(II-1) = LABEL1                                              00005420
      ICURV(II) = ICURV4                                                00005430
 9010 IF(MODCURV - 1)247,247,9015                                       00005440
  247 CALL IREADY (IDUMMY)                                              00005450
      IF(IDUMMY)5000,1260,5000                                          00005460
 1260 CALL IWRITE (ICONT, IDUMMY, 1)                                    00005470
      IF(IDUMMY)5000,260,5000                                           00005480
  260 CALL IWRITE (JXTIT, IDUMMY,11)                                    00005490
      IF(IDUMMY)5000,261,5000                                           00005500
  261 CALL IWRITE (JYTIT, IDUMMY,11)                                    00005510
      IF(IDUMMY)5000,265,5000                                           00005520
  265 IF(IT1)9269,9268,9269                                             00005530
 9268 CALL IWRITE (LTITLE, IDUMMY, 7)                                   00005540
      IF(IDUMMY)5000,9269,5000                                          00005550
 9269 IF(IT2)9271,9270,9271                                             00005560
 9270 CALL IWRITE (LTITLE(7), IDUMMY, 7)                                00005570
      IF(IDUMMY)5000,9271,5000                                          00005580
 9271 CALL IWRITE (KAXIS, IDUMMY, 5)                                    00005590
      IF(IDUMMY)5000,9015,5000                                          00005600
 9015 IF(IPOINT)9020,270,9020                                           00005610
  270 CALL IWRITE (ICURV, IDUMMY, II)                                   00005620
      IF(IDUMMY)5000,9020,5000                                          00005630
 9020 IF(MODCURV - 1)272,272,9025                                       00005640
  272 IF(IGRID - 1)9025,273,9025                                        00005650
C         GENERATE GRID IF CALLED FOR.                                  00005660
  273 IX100 = ISIZEX*100                                                00005670
      IY100 = ISIZEY*100                                                00005680
      NEXT1 = IBOTMGN                                                   00005690
      NEXT2 = LFTMGN + IX100                                            00005700
```

```
      JGRID(1) = 0                                                        00005710
      DO 1274 J=1,19,2                                                    00005720
      JGRID(J+1) = IPACK12 (LFTMGN, NEXT1, NEXT2, NEXT1)                  00005730
      IF(NEXT1 - IBOTMGN - IY100)1273,1275,1275                           00005740
 1273 NEXT1 = NEXT1 + 100                                                 00005750
      JGRID(J+2) = IPACK12 (NEXT2, NEXT1, LFTMGN, NEXT1)                  00005760
      IF(NEXT1 - IBOTMGN - IY100)1274,1276,1276                           00005770
 1274 NEXT1 = NEXT1 + 100                                                 00005780
 1275 JGRID(J+2) = IPACK12 (NEXT2, NEXT1, NEXT2, NEXT1)                   00005790
 1276 JGRID(J+3) = ICURV3                                                 00005800
      JGRID(J+4) = ICURV4                                                 00005810
      CALL IWRITE(JGRID, IDUMMY, J+4)                                     00005820
      IF(IDUMMY)5000,1277,5000                                            00005830
 1277 NEXT1 = LFTMGN                                                      00005840
      NEXT2 = IBOTMGN + IY100                                             00005850
      DO 1279 J=1,11,2                                                    00005860
      JGRID(J+1) = IPACK12 (NEXT1, IBOTMGN, NEXT1, NEXT2)                 00005870
      IF(NEXT1 - LFTMGN - IX100)1278,1280,1280                            00005880
 1278 NEXT1 = NEXT1 + 100                                                 00005890
      JGRID(J+2) = IPACK12 (NEXT1, NEXT2, NEXT1, IBOTMGN)                 00005900
      IF(NEXT1 - LFTMGN - IX100)1279,1281,1281                            00005910
 1279 NEXT1 = NEXT1 + 100                                                 00005920
 1280 JGRID(J+2) = IPACK12 (NEXT1, NEXT2, NEXT1, NEXT2)                   00005930
 1281 JGRID(J+3) = ICURV3                                                 00005940
      JGRID(J+4) = ICURV4                                                 00005950
      CALL IWRITE (JGRID, IDUMMY, J+4)                                    00005960
      IF(IDUMMY)5000,9025,5000                                            00005970
 9025 IF(IPOINT)9030,276,9030                                             00005980
C          GENERATE POINT PLOT RECORDS IF CALLED FOR.                    00005990
 9030 IOUT = 0                                                            00006000
      CALL IPACKL1 (LABEL, LABEL1, LABEL2)                                00006010
      DO 9050 I=1,NUMPTS                                                  00006020
      C1 = X(I)*SCX + SHIFTX                                              00006030
      C2 = Y(I)*SCY + SHIFTY                                              00006040
      IF(C1 - EXSIZE)9031,9031,9034                                       00006050
 9031 IF(C2 - YSIZE)9032,9032,9034                                        00006060
 9032 IF(C1 - SIZEX)9034,9033,9033                                        00006070
 9033 IF(C2 - SIZEY)9034,9035,9035                                        00006080
 9034 IOUT = IOUT +1                                                      00006090
      GO TO 9050                                                          00006100
 9035 IC1 = C1                                                            00006110
      GO TO (9036,9037,9038,9039,9040),IPOINT                             00006130
C          GENERATE CROSS.                                               00006140
 9036 ICURV(2) = IPACK12 (IC1-5, IC2-5, IC1+5, IC2+5)                     00006150
      ICURV(3) = IPACK12 (IC1  , IC2  , IC1-5, +C2+5)                     00006160
      ICURV(4) = IPACK12 (IC1+5, IC2-5, IC1+5, IC2-5)                     00006170
      GO TO 9041                                                          00006180
C          GENERATE PLUS.                                                00006190
 9037 ICURV(2) = IPACK12 (IC1  , IC2-5, IC1  , IC2+5)                     00006200
      ICURV(3) = IPACK12 (IC1  , IC2  , IC1-5, IC2  )                     00006210
      ICURV(4) = IPACK12 (IC1+5, IC2  , IC1+5, IC2  )                     00006220
      GO TO 9041                                                          00006230
C          GENERATE SQUARE.                                              00006240
 9038 ICURV(2) = IPACK12 (IC1+4, IC2-4, IC1+4, IC2+4)                     00006250
      ICURV(3) = IPACK12 (IC1-4, IC2+4, IC1-4, IC2-4)                     00006260
      ICURV(4) = IPACK12 (IC1+4, IC2-4, IC1+4, IC2-4)                     00006270
```

```
      GO TO 9041                                                        00006280
C         GENERATE DIAMOND.                                             00006290
 9039 ICURV(2) = IPACK12 (IC1+5, IC2  , IC1  , IC2+5)                   00006300
      ICURV(3) = IPACK12 (IC1-5, IC2  , IC1  , IC2-5)                   00006310
      ICURV(4) = IPACK12 (IC1+5, IC2  , IC1+5, IC2  )                   00006320
      GO TO 9041                                                        00006330
C         GENERATE TRIANGLE.                                            00006340
 9040 ICURV(2) = IPACK12 (IC1+5, IC2-3, IC1  , IC2+6)                   00006350
      ICURV(3) = IPACK12 (IC1-5, IC2-3, IC1+5, IC2-3)                   00006360
      ICURV(4) = ICURV(3)                                               00006370
 9041 IF(I - NUMPTS)9043,9042,9043                                      00006380
 9042 ICURV(5) = LABEL2                                                 00006390
      GO TO 9046                                                        00006400
 9043 IF(I - 1)9045,9044,9045                                           00006410
 9044 ICURV(5) = LABEL1                                                 00006420
      GO TO 9046                                                        00006430
 9045 ICURV(5) = ICURV3                                                 00006440
 9046 ICURV(6) = ICURV4                                                 00006450
      CALL IWRITE (ICURV, IDUMMY, 6)                                    00006460
      IF(IDUMMY)5000,9050,5000                                          00006470
 9050 CONTINUE                                                          00006480
      IF(IOUT)9048,276,9048                                             00006490
 9048 PRINT 9104, IOUT                                                  00006500
 9104 FORMAT (/, 1X, I2, 29H POINT(S) WERE OFF THE GRAPH. ,/)           00006510
C         SET UP RETURN.                                                00006520
  276 IF(MODCURV)277,278,277                                            00006530
  277 IF(MODCURV - 3)279,278,279                                        00006540
  278 ITEST = 0                                                         00006550
      PRINT 130,(JJTITLE(I),I=1,12)                                     00006560
  130 FORMAT (/, 19H GRAPH TITLED . .  ,6A8,/,19X,6A8,                  00006570
     1           24H  . . HAS BEEN PLOTTED. ,/,1H0)                     00006580
      IDUMMY = ITYP2(IDUMMY)                                            00006590
      IF(IDUMMY)5670,656,5670                                           00006600
  656 LAST = 0                                                          00006610
      RETURN                                                            00006620
  279 ITEST = 1                                                         00006630
      IDUMMY = ICLOCK(IDUMMY)                                           00006640
      LAST = 0                                                          00006650
      RETURN                                                            00006660
C         THESE ARE THE NORMAL RETURNS.                                 00006670
C         NOW SET UP THE RETURN FOLLOWING A TAPE ERROR.                 00006680
 5000 IF(MODCURV - 1)5001,5001,5002                                     00006690
 5001 IDUMMY = ITYPE1(IDUMMY)                                           00006700
      GO TO 247                                                         00006710
 5002 PRINT 5100                                                        00006720
 5100 FORMAT (/, 36H TAPE ERROR IN WRITING GRAPH OUTPUT. )              00006730
      IDUMMY = ITYPE1(IDUMMY)                                           00006740
      GO TO 1007                                                        00006750
 5670 IDUMMY = ITYPE1(IDUMMY)                                           00006760
      END                                                               00006770
C                                                                       00006780
C                                                                       00006790
      SUBROUTINE IREADY (IDUMMY)                                        00006800
C         SELECTS TAPE 8 (. LL LOOP UNTIL READY). WRITES EOF ON 8.      00006810
*         MACHINE LANGUAGE WILL NOT BE NECESSARY IN FORTRAN 62-3.       00006820
      LOC(IFIVE = 5).                                                   00006830
```

```
        EXF (520418)     EXF7 (52000B).   SELECT READ AND WAIT TAPE.      00006840
 1NEX   EXF7(00050B)     SLJ (1RDY).      EXIT ON CH 5 ACTIVE.           00006850
        EXF7(52000B)     SLJ (1NEX).      EXIT ON TAPE READY.            00006860
        LDA (IFIVE)      SAU (1BUF).      TERMINATE                      00006870
 1BUF   EXF5(N).                            BUFFER.                      00006880
 1RDY   EXF (52041B)     EXF7(52000B).    SELECT AND WAIT TAPE 8.        00006890
        ENA (0).                         CLEAR A.                       00006900
        EXF (02000B)     EXF (52006B).    STOP CLOCK AND BACKSPACE 8.    00006910
        EXF7(52001B)     SLJ (1END).      EXIT IF NOT AT LOAD POINT.     00006920
       -EXF7(52000B).                    WAIT TAPE 8.                   00006930
        EXF7(52007B)     SLJ (1EOF).      EXIT IF NO EOF.                00006940
        ENA (IDUMMY)     SAU (2BUF).      MOVE                           00006950
        INA (1)          STA (IFIVE).       FORWARD                      00006960
 2BUF   EXF5(N)          EXF7(52000B).        OVER RECORD.               00006970
 1EOF   ENA (0)          EXF7(00061B).    CLEAR A, WAIT CH 6.            00006980
        EXF (62041B)     EXF7(62000B).    SELECT WRITE AND WAIT TAPE.    00006990
        EXF (62041B)     EXF7(62000B).    SELECT AND WAIT TAPE 8.        00007000
        EXF (62003B)     EXF7(62000B).    WRITE EOF AND WAIT.            00007010
        EXF7(62007B)     ENA(10).         EXIT ON NO END OF TAPE.        00007020
 1END   STA (IDUMMY).                                                   00007030
      END                                                               00007040
C                                                                       00007050
      SUBROUTINE IWRITE(ISTART, IDUMMY, IWORDS)                         00007060
C        WRITE RECORD OF IWORDS, STARTING WITH ISTART. PUT IDUMMY = 0   00007070
C         IF RECORD CORRECTLY WRITTEN, OTHERWISE SET NON-ZERO.          00007080
*         MACHINE LANGUAGE WILL NOT BE NECESSARY IN FORTRAN 62-3.       00007090
      LOC(ISIX = 6).                                                    00007100
      -EXF7(00061B).                   WAIT CH 6.                     00007110
        EXF (62041B)     EXF7(62000B).    SELECT WRITE, WAIT TAPE.       00007120
        EXF (62041B)     EXF7(62000B).    SELECT AND WAIT TAPE 8.        00007130
        ENQ (111B).                       SET COUNTER.                  00007140
 1AGN   ENA (ISTART)     INA(1).          STARTING ADDRESS.              00007150
        SAL (1BUF)       ADD(IWORDS).     TERMINAL ADDRESS.              00007160
 1BUF   STA(ISIX)        EXF6(N).         BUFFER OUT.                    00007170
        ENA(0)           EXF7(62000B).    CLEAR A. WAIT TAPE 8.          00007180
        EXF7(62007B)     SLJ (1END).      EXIT IF NO END OF TAPE.        00007190
        EXF7(62003B)     SLJ (2AGN).      EXIT IF NO PARITY ERROR.       00007200
        EXF7(62004B)     SLJ (2END).      EXIT IF LENGTH ERROR.          00007210
 2AGN   EXF (62006B)     EXF7(62000B).    BACKSPACE AND WAIT.            00007220
        QRS (3)          QJP1(1AGN).      TRY WRITE 3 TIMES.             00007230
 1END   EXF (62003B)     ENA (10).        WRITE EOF, NON ZERO A.         00007240
 2END   STA (IDUMMY).                     STORE RESPONSE.                00007250
      END                                                               00007260
C                                                                       00007270
      FUNCTION ITYP2 (IDUMMY)                                           00007280
C         TYPE WORD GRAPH.                                              00007290
*         WILL NEED REWRITING IN FORTRAN 62-3.                          00007300
      CON(LC = 57B, M1 = 4513123015050000B ).                           00007310
      LOC(ITWO = 2).                                                    00007320
      -EXF7(00061B).                   WAIT CH 6.                     00007330
        EXF (62041B)     EXF7(62000B).    SELECT AND WAIT TAPE.          00007340
        EXF (62041B)     EXF7(62000B).    SELECT AND WAIT TAPE 8.        00007350
        EXF (62003B)     EXF (01000B).    WRITE EOF. START CLOCK.        00007360
        ENA (0)          EXF7(62000B).    CLEAR A. WAIT TAPE 8.          00007370
        EXF7(62007B)     ENA (10).        EXIT IF NO END OF TAPE.        00007380
        STA (ITYP2).                      STORE RESPONSE.                00007390
```

```
      -EXF7(00021B).                        WAIT CH 2.                                00007400
       EXF7(11141B)     SLJ (1TYP).         EXIT IF UPPER CASE.                       00007410
       EXF (21100B)     ENA (LC+1).         TYPE                                      00007420
       STA (ITWO)       EXF2(LC).             LOWER CASE.                             00007430
      -EXF7(00021B).                        WAIT CH 2.                                00007440
 1TYP  EXF (21100B)     ENA (M1+1).         TYPE                                      00007450
       STA (ITWO)       EXF2(M1).             GRAPH                                   00007460
      END                                                                            00007470
C                                                                                    00007480
      FUNCTION ITYPE1 (IDUMMY)                                                       00007490
C         REWIND TAPE 8, REQUEST NEW TAPE, AND WAIT TILL READY.                      00007500
*         WILL NEED REWRITING IN FORTRAN 62-3.                                       00007510
      CON(LC = 57B, M1 = 4515112030242004B, M2 = 1103302204062031B,                  00007520
     1    M3 = 0401301520043342B).                                                   00007530
      RSV(MESS = 3).                                                                 00007540
      LOC(ITWO = 2).                                                                 00007550
      -EXF7(00061B).                        WAIT CH 6.                               00007560
       EXF (62041B)     EXF7(62000B).       SELECT AND WAIT TAPE.                    00007570
       EXF (62041B)     EXF7(62000B).       SELECT AND WAIT TAPE 8.                  00007580
       EXF (62003B)     EXF7(62000B).       WRITE EOF AND WAIT.                      00007590
       EXF (62007B)     EXF7(00021B).       REWIND WITH INTERLOCK, WAIT CH 200007600
       EXF7(11141B)     SLJ (1TYP).         EXIT IF UPPER CASE.                      00007610
       EXF (21100B)     ENA (LC+1).         TYPE                                     00007620
       STA (ITWO)       EXF2(LC).             LOWER CASE.                            00007630
 1TYP  LDA (M1)         STA (MESS).                                                  00007640
       LDA (M2)         STA (MESS+1).                                                00007650
       LDA (M3)         STA (MESS+2).                                                00007660
      -EXF7(00021B).                        WAIT CH 2.                               00007670
       EXF (21100B)     ENA (MESS+3).       TYPE                                     00007680
       STA (ITWO)       EXF2(MESS).           MESSAGE.                               00007690
      -EXF7(00061B).                        WAIT CH 6.                               00007700
      -EXF7(62000B).                        WAIT TAPE.                               00007710
       EXF (62041B)     EXF7(62000B).       SELECT AND WAIT TAPE 8.                  00007720
       EXF (01000B).                        START CLOCK.                             00007730
      END                                                                            00007740
C                                                                                    00007750
      FUNCTION ICODE (ANUMBER)                                                       00007760
C         CODES ABSOLUTE VALUE OF A FLOATING POINT NUMBER (BETWEEN                   00007770
C         1.0E-100 AND 1.0E+100) INTO 8-CHARACTER BCD WORD OF THE FORM               00007780
C         1.23E+45. ICODE = 8H0.00E+00 IF MAGNITUDE OUT OF RANGE.                    00007790
*         CHECK AVAILABILITY OF LIBRARY FUNCTIONS IN FORTRAN 62-3.                   00007800
      DIMENSION II(8)                                                                00007810
      BNUMBER = ABSF(ANUMBER)                                                        00007820
      IF(BNUMBER - 1.0E+100)7,6,6                                                    00007830
    7 IF(BNUMBER - 1.0E-100)6,6,2                                                    00007840
    6 ICODE = 8H0.00E+00                                                             00007850
      RETURN                                                                         00007860
C         THIS IS ERROR EXIT.                                                        00007870
    2 CALL SCALEIT (BNUMBER, ISCAL10, FACTOR, 3)                                     00007880
      ISIGNSC = XSIGNF(1,ISCAL10)                                                    00007890
      ISCAL10 = XABSF(ISCAL10)                                                       00007900
      IFACT = FACTOR*100.001                                                         00007910
      II(8) = XMODF(ISCAL10,10)                                                      00007920
      II(7)=ISCAL10/10                                                               00007930
      IF(ISIGNSC)4,3,3                                                               00007940
    3 II(6) = 8H       +                                                           00007950
```

```
      GO TO 5                                                          00007960
    4 II(6) = 8H        -                                              00007970
    5 II(5) = 8H        E                                              00007980
      II(4) = XMODF(IFACT,10)                                          00007990
      II(3) = (XMODF(IFACT,100))/10                                    00008000
      II(2) = 8H        .                                              00008010
      II(1) = IFACT/100                                                00008020
      CALL IPACK (II, IPACKED)                                         00008030
      ICODE = IPACKED                                                  00008040
      RETURN                                                           00008050
      END                                                              00008060
C                                                                      00008070
      SUBROUTINE SCALEIT (ANUMBER,ISCAL10,FACTOR,MODE)                 00008080
C         FINDS FACTOR (BETWEEN 1.0 AND 9.99...) AND SCALE OF 10 AS    00008090
C         DEFINED BY     ANUMBER = FACTOR*10.**ISCAL10.                00008100
C         MODE IS THE NUMBER OF SIGNIFICANT FIGURES REQUIRED. THIS MUST 00008110
C         BE BETWEEN 1 AND 10 OR IT WILL BE PUT EQUAL TO SIX.          00008120
*         CHECK AVAILABILITY OF LOG10F IN FORTRAN 62-3.                00008130
      ISCAL10=LOG10F(ANUMBER)                                          00008140
      FACTOR = ANUMBER/10.**ISCAL10                                    00008150
      IF(FACTOR - 0.1)1,2,2                                            00008160
    1 FACTOR = FACTOR*100.                                             00008170
      ISCAL10 = ISCAL10 - 2                                            00008180
      GO TO 8                                                          00008190
    2 IF(FACTOR - 1.0)3,8,4                                            00008200
    3 FACTOR = FACTOR*10.                                              00008210
      ISCAL10 = ISCAL10 - 1                                            00008220
      GO TO 8                                                          00008230
    4 IF(FACTOR - 100.0)6,5,5                                          00008240
    5 FACTOR = FACTOR/100.                                             00008250
      ISCAL10 = ISCAL10 + 2                                            00008260
      GO TO 8                                                          00008270
    6 IF(FACTOR - 10.0)8,7,7                                           00008280
    7 FACTOR = FACTOR/10.                                              00008290
      ISCAL10 = ISCAL10 + 1                                            00008300
    8 IF(MODE)9,9,10                                                   00008310
    9 MODE = 6                                                         00008320
      GO TO 11                                                         00008330
   10 IF(MODE - 10)11,11,9                                             00008340
   11 IFACTOR = FACTOR*10.**(MODE - 1) + 0.5                           00008350
      FACTOR = IFACTOR                                                 00008360
      FACTOR = FACTOR/10.**(MODE - 1)                                  00008370
      IF(FACTOR - 10.)13,12,12                                         00008380
   12 FACTOR = 1.                                                      00008390
      ISCAL10 = ISCAL10 + 1                                            00008400
   13 RETURN                                                           00008410
      END                                                              00008420
C                                                                      00008430
      SUBROUTINE IPACK (II, IPACKED)                                   00008440
C         TAKES 8 SIX-BIT WORDS AND PACKS THEM LEFT TO RIGHT           00008450
C         IN IPACKED. IF WORD IS ZERO, 12B IS SUBSTITUTED.             00008460
*         CONVERT TO CODAP FOR FORTRAN 62-3.                           00008470
      CON(IZERO = 12B).                                                00008480
       SIU1(ISAVE)      ENI1(8).                                       00008490
 1NEX  LDA1(II)         AJP1(2NEX).                                    00008500
       LDA (IZERO).                                                    00008510
```

```
 2NEX  LRS (6)          INI1(-2).                                        00008520
       ISK1(-1)         SLJ (1NEX).                                      00008530
       STQ (IPACKED)    LIU1(ISAVE).                                     00008540
      END                                                               00008550
C                                                                       00008560
      SUBROUTINE ISHIFT6 (ITITLE, LTITLE)                               00008570
C         INSERTS 02B AHEAD OF 6-WORD TITLE RECORD.                     00008580
*        WILL HAVE TO BE CONVERTED TO CODAP IN FORTRAN 62-3.            00008590
*         WATCH ARRAY INDEXING IN FORTRAN 62-3.                         00008600
      CON(IBLANK = 2020202020202020B).                                  00008610
       SIU1(ISAVE)     ENI1(1).          SAVE INDEX, SET COUNT.         00008620
       ENA (2).                          ENTER 02B.                     00008630
 1NEX  LDQ1(ITITLE)    LLS (42).         PERFORM                        00008640
       STA1(LTITLE)    LLS (6).            SHIFTING.                    00008650
       ISK1(6)         SLJ (1NEX).       CHECK IF COMPLETE.             00008660
       LDQ (IBLANK)    LLS (42).         COMPLETE LAST WORD.            00008670
       ENI1(7)         STA1(LTITLE).     STORE LAST.                    00008680
       ENA (2).                          REPEAT                         00008690
 2NEX  LDQ1(ITITLE)    LLS (42).           FOR                          00008700
       INI1(1)         STA1(LTITLE).         SECOND                     00008710
       INI1(-1)        LLS (6).               TITLE                     00008720
       ISK1(12)        SLJ (2NEX).               LINE.                  00008730
       LDQ (IBLANK)    LLS (42).                                        00008740
       ENI1(14)        STA1(LTITLE).                                    00008750
       LIU1(ISAVE).                      RESTORE INDEX.                 00008760
      END                                                               00008770
C                                                                       00008780
      FUNCTION IPACK12 (IONE,I2,I3,I4)                                  00008790
C         PACKS FOUR 12-BIT WORDS INTO ONE 48-BIT WORD.                 00008800
*         WILL REQUIRE CONVERSION TO CODAP IN FORTRAN 62-3.             00008810
      LDA(IONE)      LDQ(I2).                                           00008820
      QLS(36)        LLS(12).                                           00008830
      LDQ(I3)        QLS(36).                                           00008840
      LLS(12)        LDQ(I4).                                           00008850
      QLS(36)        LLS(12).                                           00008860
      STA(IPACK12).                                                     00008870
      END                                                               00008880
C                                                                       00008890
      SUBROUTINE IPACKL1 (LABEL, LABEL1, LABEL2)                        00008900
C         PACKS TWO 4-CHARACTER LABELS.                                 00008910
*         USE DECODE/ENCODE IN FORTRAN 62-3.                            00008920
      CON(IFLAG = 37773777B).                                           00008930
       LLS(24)          STA (LABEL1).                                   00008950
       LDA (IFLAG)      LDQ (LABEL).                                    00008940
       LDA (IFLAG)      LLS(24).                                        00008960
       STA (LABEL2).                                                    00008970
      END                                                               00008980
C                                                                       00008990
      FUNCTION ICLOCK(IDUMMY)                                           00009000
       EXF (01000B).                    START CLOCK.                    00009010
      END                                                               00009020
      IC2 = C2                                                          00006120
      END                                                               00009030
      END
```

APPENDIX B

PROGRAM NOMENCLATURE

The following listing of program symbols is provided to help the reader understand symbols used in the digital computer program. Symbols identified in section 3 as well as symbols readily identifiable are not listed in this appendix.

X(Y)	individual strain gage readings
Y(I)	percentage of depth of individual strain gage readings
AZ	the zero depth intercept of strain axis
ANG(L)	the angle of σ_{max} from gage leg number 1
SMAX(L)	σ_{max}
SMIN(L)	σ_{min}
THETA(L)	θ_p
SIGXX(LL)	experimental value of σ_N
TRUXY(L)	experimental value of τ_{xy}
PHI(L)	the angle ϕ
XSXX(L) YSXX(L)	components of experimental values of σ_n in proper form for plotting
XTXY(L) XTXY(L)	components of experimental values of τ_{xy} in proper form for plotting
ZX(1,1) ↓ ZX(7,1)	seven coefficients used in computing smoothed value of σ_N
XY(1,1) ↓ XY(7,1)	seven coefficients used in computing smoothed value of τ_{xy}
ZZZ	the angle in radians on the circumference of the piping between two consecutive smoothed values of σ_N or τ_{xy}
PHI 1(1)	the angle in radians from rosette number 12 of a smoothed value of σ_N or τ_{xy}
SIGN 7(I)	smoothed value of σ_N.

TAUN 7(I) smoothed value of τ_{xy}

XSNR(I), YSNR(I) } components for plotting zero stress circle on σ_N plot

XTNR(I), YTNR(I) } components for plotting zero stress circle on τ_{xy} plot

XSN7(I), YSN7(I) } components of smoothed value of σ_N in proper form for plotting

XTN7(I), YTN7(I) } components of smoothed value of τ_{xy} in proper form for plotting

APPENDIX C

DESCRIPTION OF SUBROUTINE GAUSS 2

A. IDENTIFICATION

TITLE: Solution of Simultaneous Linear Algebraic Equations

CO-OP ID: F2

FORTRAN: Gauss 2

CATEGORY: Simultaneous Linear Equations

PROGRAMMER: C. B. Bailey and Mary Haynes

DATE OF COMPLETION: April 1963

B. PURPOSE: To solve one or more sets of linear algebraic equations using Gaussian elimination with row pivoting and back substitution.

C. USAGE:

1. Calling sequence

Call Gauss 2 (N, M, EP, A, B, X, KER)

N - order of the matrix in the equation Ax = b, that is, the number of linear equations (maximum = 50).

M - number of vectors b for which solution vectors x are to be obtained, that is, the number of sets of linear equations (Maximum = 60).

EP - matrix condition parameter (see Mathematical Method).

A - the elements of the matrix of coefficients of the equations, stored in the form (A_{ij})

$i = 1, 2, \ldots, N$

$j = 1, 2, \ldots, N$

A is dimensional as a 50 x 50 array.

B - the elements of the column vectors b etc., stored in the form (B_{ij})

$i = 1, 2, \ldots, N$

$j = 1, 2, \ldots, M$

B is dimensional as a 50 x 60 array.

X - the components of the x vectors stored in the form (X_{ij}).

$i = 1, 2, \ldots, N$

$j = 1, 2, \ldots, M$

X is dimensional as a 50 x 60 array.

KER - error flag (See Error Return)

2. Space Required:

3. Temporary or Common Storage Required

4. Error Return: Argument KER

   1. indicates no errors

   2. indicates that matrix is singular or nearly singular.

5. Accuracy: Not applicable.

D. METHOD

This subroutine is an adaptation of the program F2 VTEX LINEQN. Please consult the write-up of that program for the mathematical method. The subroutine gives no printout.

APPENDIX D

DESCRIPTION OF SUBROUTINE DRAW

A. IDENTIFICATION:

TITLE: General Graph Output Subroutine

CO-OP ID: J7-NPS-DRAW

CATEGORY: Output for Off-Line Plotting

PROGRAMMER: J. R. Ward

DATE: February 1964; REVISED JUNE 1965

B. PURPOSE:

This subroutine, when provided with the necessary information, generates a magnetic tape in the proper format for subsequent off-line graph plotting using the CDC 160 or 160A GRAFPLOT program and a CalComp 165 Plotter (see references 1 and 2). Provision is made for curve drawing and point plotting, automatic scaling, graph titling and axis annotating. An attempt was made to provide a considerable amount of flexibility, at the expense, necessarily, of a relatively large number of arguments and a rather high memory requirement.

C. USAGE:

1. Definitions:

In what follows the word "graph" will be taken to mean one piece or frame of graph paper on which there may be plotted one or more curves and/or sets of points. A "curve" will mean a continuous line generated by the sequence of straight lines joining successive points of the set defining the curve. A "point plot" will describe the representation of a succession of points by means of symbols (such as a cross) on the graph. The points are not connected in a point plot.

2. Calling Arguments:

All necessary information is transferred to DRAW through the calling

arguments. The call statement is: CALL DRAW (NUMPTS,X,Y,MODCURV, ITYPE,LABEL,ITITLE,EXSCALE,YSCALE,IXUP,IYRIGHT,MODEXAX,MODEYAX,IWIDE, IHIGH,IGRID,LAST)

It is important to realize that one and only one curve or set of points is plotted each time DRAW is called. However, it is possible to call DRAW repeatedly if several curves and/or sets of points are wanted on one graph.

The calling arguments are as follows:

a. NUMPTS: The number of points defining a curve ($2 \leq$ NUMPTS ≤ 900), or the number of points to be point plotted ($2 \leq$ NUMPTS ≤ 30).

b. X: The array of X-ordinates ($|X_i| \leq 10^{99}$ for i=1, 2,..., NUMPTS). X must be dimensioned at least equal to NUMPTS in the calling program.

All points will be considered to have the same X-ordinate if $|X_{max} - X_{min}| < 10^{-97}$. The common value will be put equal to zero if $|X_{max}| \leq 10^{-97}$.

c. Y: The array of Y-ordinates, with properties corresponding to the X-ordinates, above. Y must be dimensioned at least equal to NUMPTS in the calling program.

d. MODCURV: Controls the number of curves, and/or sets of points on one graph:

= 0 This is the only curve, or set of points, to be plotted on this graph.

= 1 This is the first of two or more curves, and/or sets of points, to be plotted on this graph.

= 2 This is an intermediate curve, or set of points.

= 3 This is the last curve, or set of points, for this graph.

e. ITYPE: Controls the type of plot (i.e., curve or point plot):

= 0 This set of points is to be represented by a curve.

= 1 These points are to be plotted with a cross (x).

= 2 These points are to be plotted with a plus (+).

= 3 These points are to be plotted with a square (□).

= 4 These points are to be plotted with a diamond (◇).

= 5 These points are to be plotted with a triangle (△).

f. LABEL: This is a Hollerith curve or point identifier. If a curve is being drawn, LABEL must have 4 characters (including any blanks), and these will be reproduced beside the end of the curve. This argument can be set in the calling program by a statement such as

LABEL = 4H ONE ($\wedge$ = blank)

or LABEL = 4H1234

or LABEL = 4H$\wedge\wedge\wedge\wedge$ The latter must be used when no label is wanted.

If a set of points is being plotted, LABEL is an 8-character identifier. The first 4 characters will be reproduced beside the first point, and the last four characters will be reproduced alongside the last point. This argument can be set by statements such as

LABEL = 8HFRSTLAST

or LABEL = 8H$\wedge\wedge\wedge\wedge$ONE

or LABEL = 8H$\wedge$ONE$\wedge$123

or LABEL = 8H$\wedge\wedge\wedge\wedge\wedge\wedge\wedge\wedge$ The latter must be used if no labels are wanted.

The above arguments, a. through f. (and q.), have meaning every time DRAW is called. On the other hand,

the remaining arguments, g. through p., have no meaning except when MODCURV = 0 or 1.

g. ITITLE: An array of twelve 8-character Hollerith words, the first six of which will form the first title line, and the last six the second. The array must be dimensioned 12 in the calling program, must contain the user's job identification, and must have unwanted characters set to blank. For example:

```
      DO 1 I = 1,12
1     ITITLE(I) = 8H
      ITITLE(1) = 8H^SMITH,^
      ITITLE(2) = 8HJ.^J.^^^
      ITITLE(7) = 8H^TESTIT.
```

h. EXSCALE: X-scale in units per inch ($10^{-99} < \text{EXSCALE} < 10^{99}$). EXSCALE will always be rounded off to one figure significance. If EXSCALE = 0, the X-scale will be computed by DRAW. This is called auto-scale.

f. YSCALE: Y-scale in units per inch, with properties corresponing to those of EXSCALE.

j. IXUP: Distance, in inches, of the X-axis from the bottom of the graph ($0 \leq \text{IXUP} \leq \text{IHIGH}$). This argument will be ignored unless MODEXAX = 2, see below.

k. IYRIGHT: Distance, in inches, of the Y-axis from the left of the graph ($0 \leq \text{IYRIGHT} \leq \text{IWIDE}$). This will be ignored unless MODEYAX = 2, see below.

l. MODEXAX: Determines the mode of the X-axis location:

= 0 The X-axis will be located automatically by DRAW, with the origin of Y on the graph.

= 1 The X-axis will be automatically located by DRAW, with the origin of Y removed (in one's imagination) an integer number of inches above or below the graph, if this is appropriate. This option can be used only if the Y-scaling is automatic (YSCALE = 0).

= 2 The X-axis location will be as specified by IXUP.

m. MODEYAX: Determines the mode of Y-axis location in the same way as MODEXAX, above, governs the X-axis location.

n. IWIDE: Width of graph in inches ($1 \leq$ IWIDE ≤ 9). If IWIDE is out of this range, a value of 8 will be assumed.

o. IHIGH: Height of graph in inches ($1 \leq$ IHIGH ≤ 15). If IHIGH is out of this range, a value of 8 will be assumed.

p. IGRID: If IGRID = 1, a 1" x 1" grid will be superimposed on the graph. This is useful only if plain paper is used on the CalComp Plotter.

q. LAST: Indicates to the calling program whether the previous plot was completed successfully. The codes are:

= 0 Last plot was completed successfully.

= 1 Last plot was not completed successfully.

= 2 Last plot was not completed successfully, and no further graph output will be attempted until DRAW is next entered with MODCURV = 1 or 0.

= 3 An attempt was made to enter DRAW with MODCURV $\neq$ 1 or 0 while the error lockout was set.

This argument must always be a variable name and never a number in the call statement.

3. Notes and Comments:

a. The graph scales-and, if MODEXAX = 1 and/or MODEYAX = 1, the

amounts of origin offset - are always output as part of the graph title.

b. Each time a graph is completed, a message to this effect is printed on both the standard output and the console typewriter.

c. There are internal checks of the input to DRAW to prevent incorrect use. If an input error is detected, an attempt will be made, where possible, to complete the plot. If an argument is "corrected" in this process, the user will be so informed on the standard output. If it is not possible to complete the plot, the user will be informed of the reason by a message on the standard output.

d. If part or all of a curve would fall more than 0.6" laterally beyond the ends of the X-axis, or 0.7" vertically beyond the ends of the Y-axis, the X and/or Y ordinates will be limited so that the curve will typically become a line along part or all of the boundary of the graph as here defined.

e. If one or more points of a point plot would fall outside the graph area, the plot of that point, or points will be inhibited. The number of such points will be reported to the user on the standard output.

f. It should be pointed out that the X and Y scaling and axis locating processes are entirely independent, so that, for example, X might be auto-scaled, while the Y-scale is specified. At the same time the X-axis might be located automatically, while the Y-axis location is specified.

g. It must be remembered that the scales and axis locations of a multi-plot graph are set when DRAW is called for the first time (with MODCURV = 1). Thus the user must attempt, at that time, to achieve scaling and axis location which will be appropriate to all the plots he intends to make on the one graph. Particularly if the automatic features of DRAW are selected, foresight will be demanded of the user in this respect.

4. Auto-Scale Properties:

The scale factor is chosen from amongst the values 1,2, or 5 units per inch, or some power of 10 times one of those values. A curve, or set of points which is plotted with auto-scale will normally lie entirely within the graph area as defined in 3.d., above. The only exception may occur if an axis is placed, by the user, along one edge of the graph (e.g., IXUP = 0, MODEXAX = 2). In such a case, points "outside" the axis are not considered in the selection of a scale factor (e.g., negative X_i do not affect the choice of scale when IXUP = 0). If automatic axis location as well as auto-scale is selected, the plot, if it does not fill the graph area, will be placed as far as possible towards the bottom-left of the graph area consistant with the fact that the axes can be set in increments of 1" only.

5. Space Required: 3960 cells (excluding the input arrays).

6. Temporary Storage: None.

7. Error Print-Outs: There are a large number of possible error print-outs. These are all self-explanatory.

8. Error Returns: All error returns are preceded by self-explanatory error printouts. An error indication is transferred back to the calling program through the argument LAST.

9. Error Stops: None.

10. Tape Mountings: Logical Tape #8 will receive the binary graph output. The standard monitor output will receive the messages to the user.

11. Output Format: The format of the binary graph output records on magnetic tape is described in reference 1. The only difference is that in this program the interpolation option is by-passed (Set to zero in the graph output record). See reference 2.

12. Selective Jump and Stop Settings: None.

13. Timing: Variable, depending upon the number of points and the options chosen. Typically less than one second per curve or point plot.

14. Accuracy: The accuracy of results is equal to the resolution of the CalComp Plotter, that is, 0.01" in both the X and Y directions.

15. Equipment Configuration: CDC 1604 with FORTRAN 60 compiler and Library. A CDC 160 or 160A with CalComp 165 Plotter is needed for the off-line plotting using the appropriate GRAFPLOT program.

D. REFERENCES:

1. Weir, Maurice D., Spritzer, Milton and McIlhenny, D.W., "160-A Graph Plot Program," Ident *B001, SWAP Library, 15 August 1962.

2. Hogg, R. L. and Glover, D.C., "160 Grafplot Routine", Writeup Available from Computer Facility, U. S. Naval Postgraduate School, 1 April 1963.

3. U. S. Naval Postgraduate School, Thesis, "Control System Programming, Remote Computing and Data Display," by Robert Lee Hogg and Dennis C. Glover, 1963.

N.B. References 1 and 2 are included in Reference 3 as Enclosures 2 and 1, respectively.

APPENDIX E

PLOTS OF DOCK TRIAL DATA

Scale of stress has been omitted for reasons of security

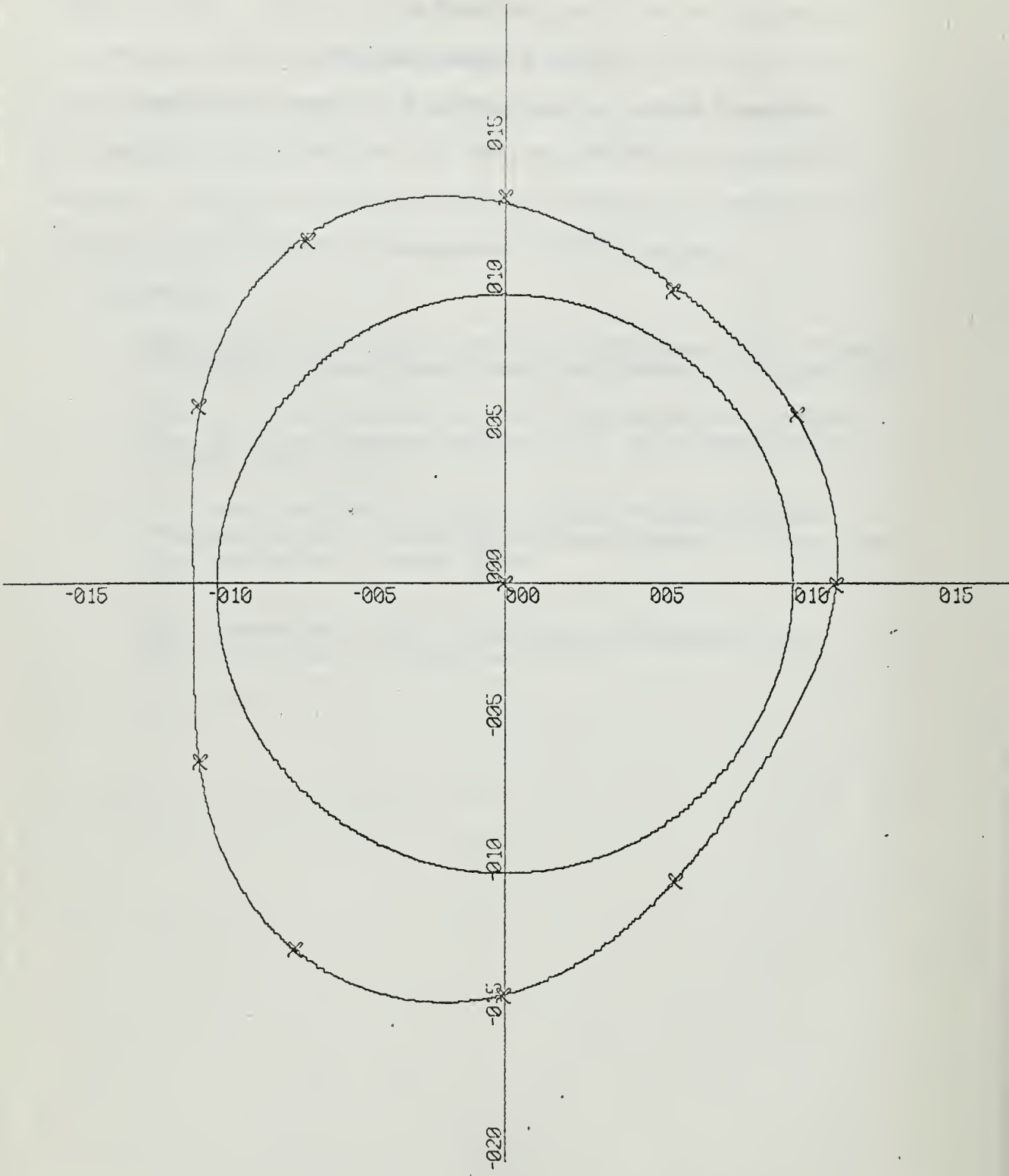


FIGURE E-1

Polar plot of σ_N for rosette ring #1

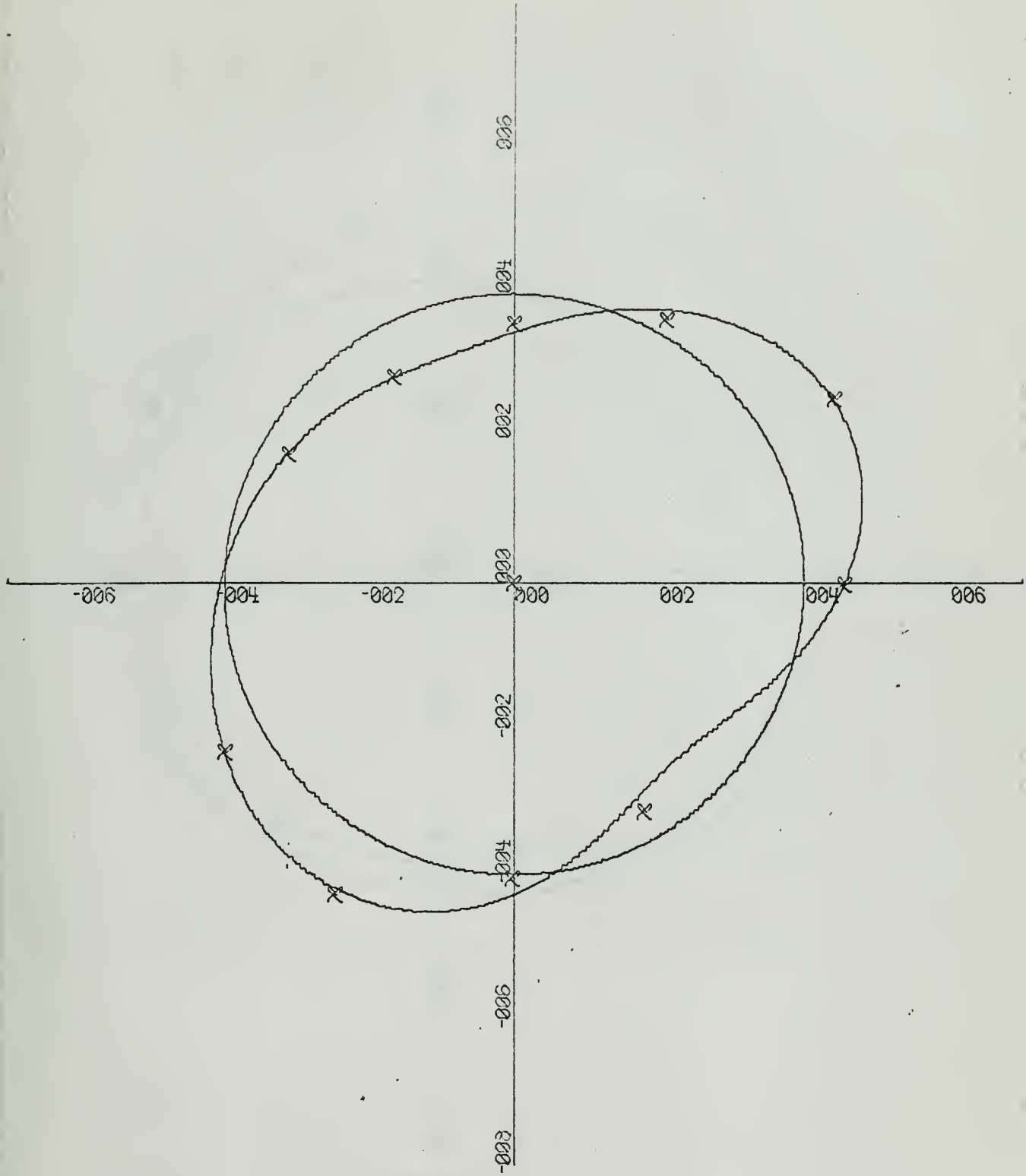


FIGURE E-2

Polar plot of τ_{xy} for rosette ring #1

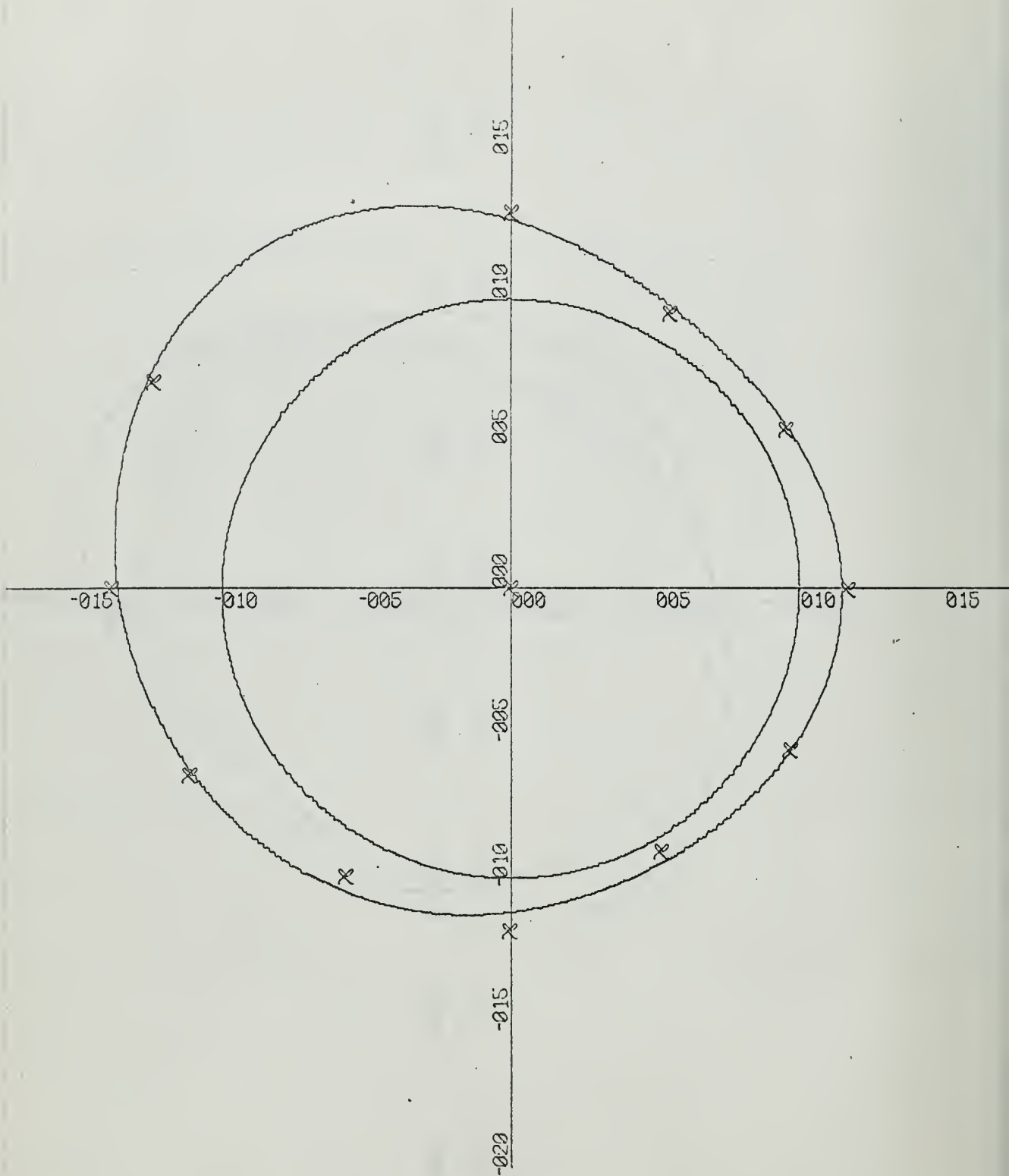


FIGURE E-3

Polar plot of σ_N for rosette ring #2

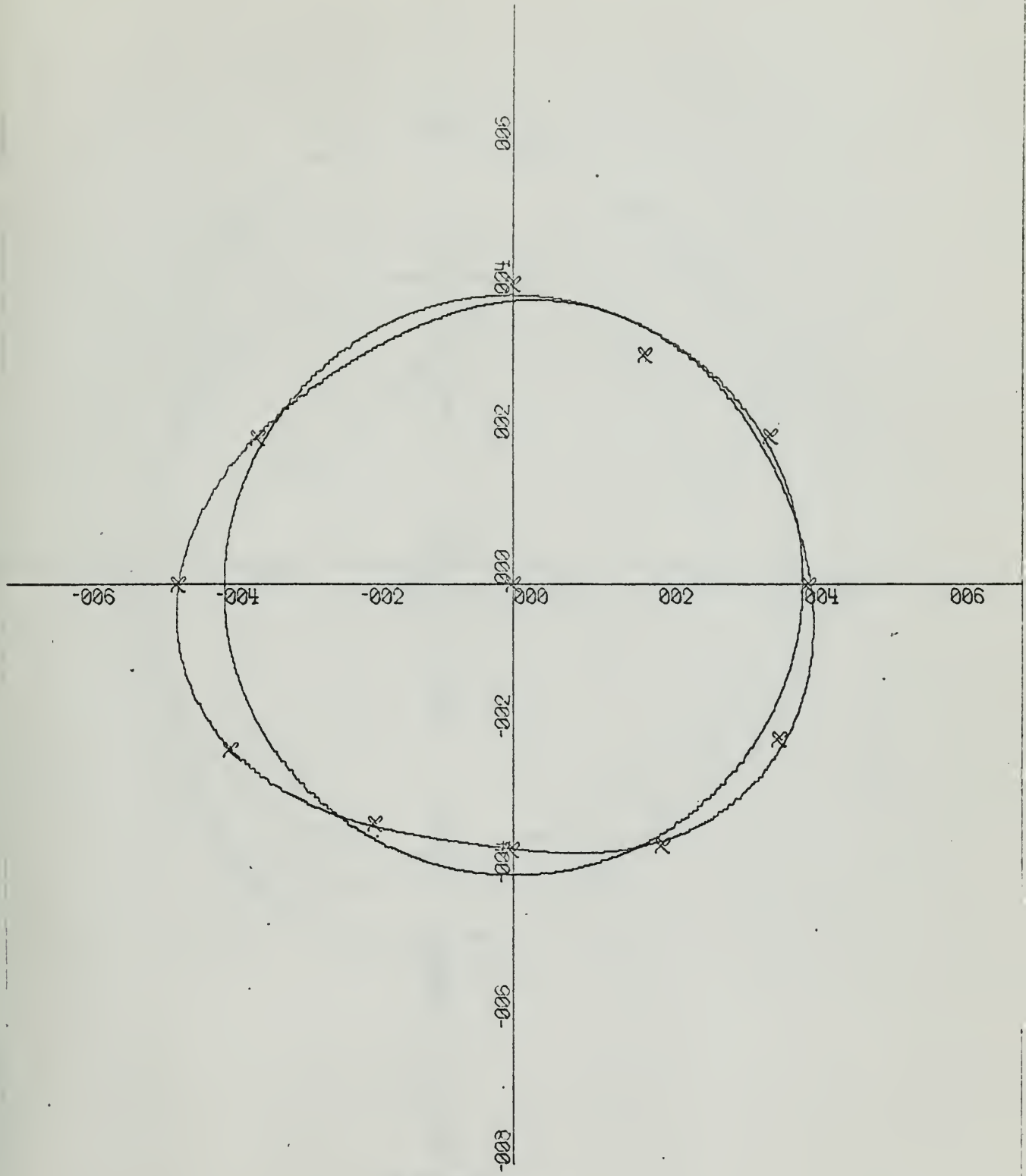


FIGURE E-4

Polar plot of τ_{xy} for rosette ring #2

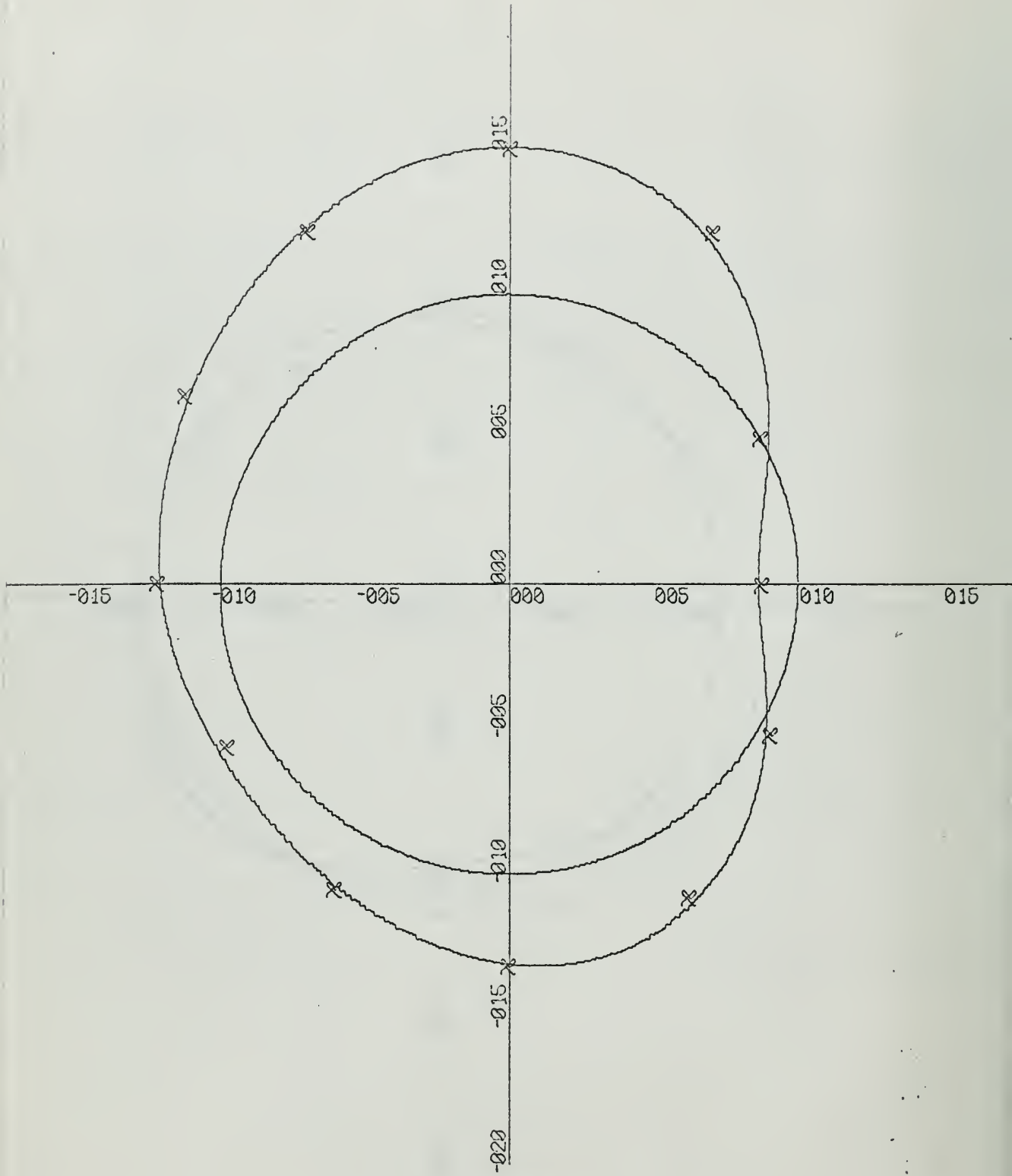


FIGURE E-5

Polar plot of σ_N for rosette ring #3

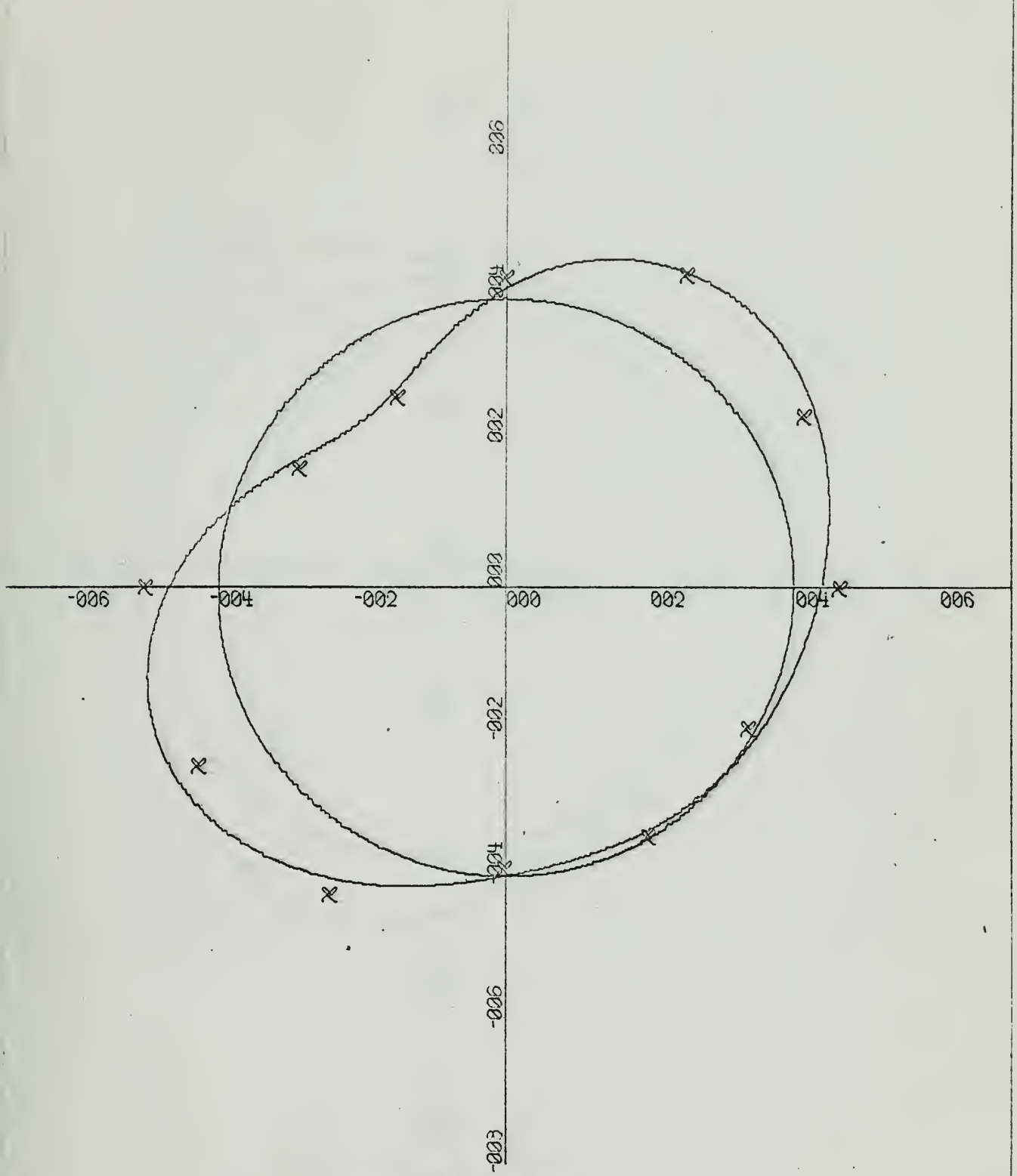


FIGURE E-6

Polar plot of τ_{xy} for rosette ring #3

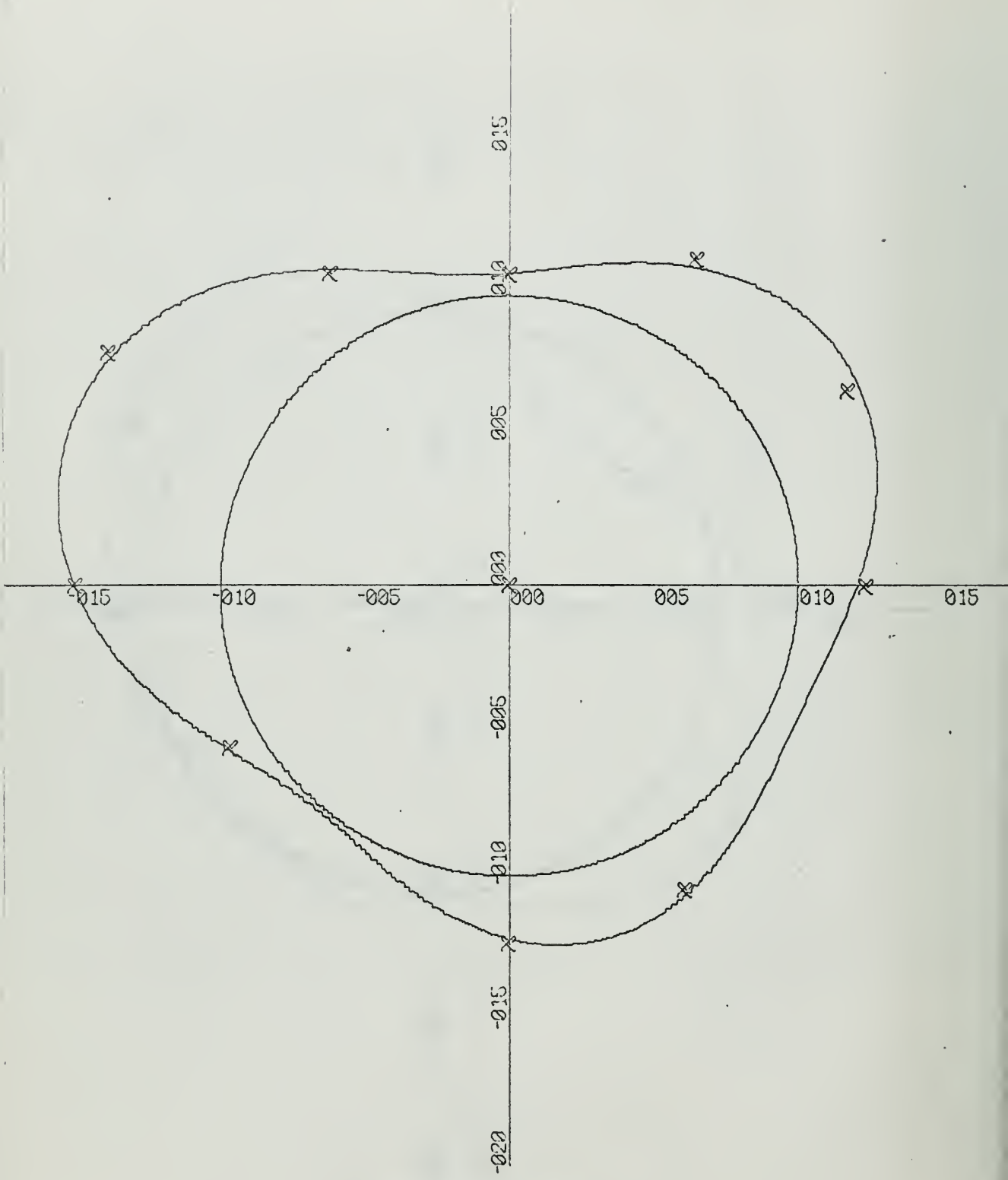


FIGURE E-7

Polar plot of σ_N for rosette ring #4

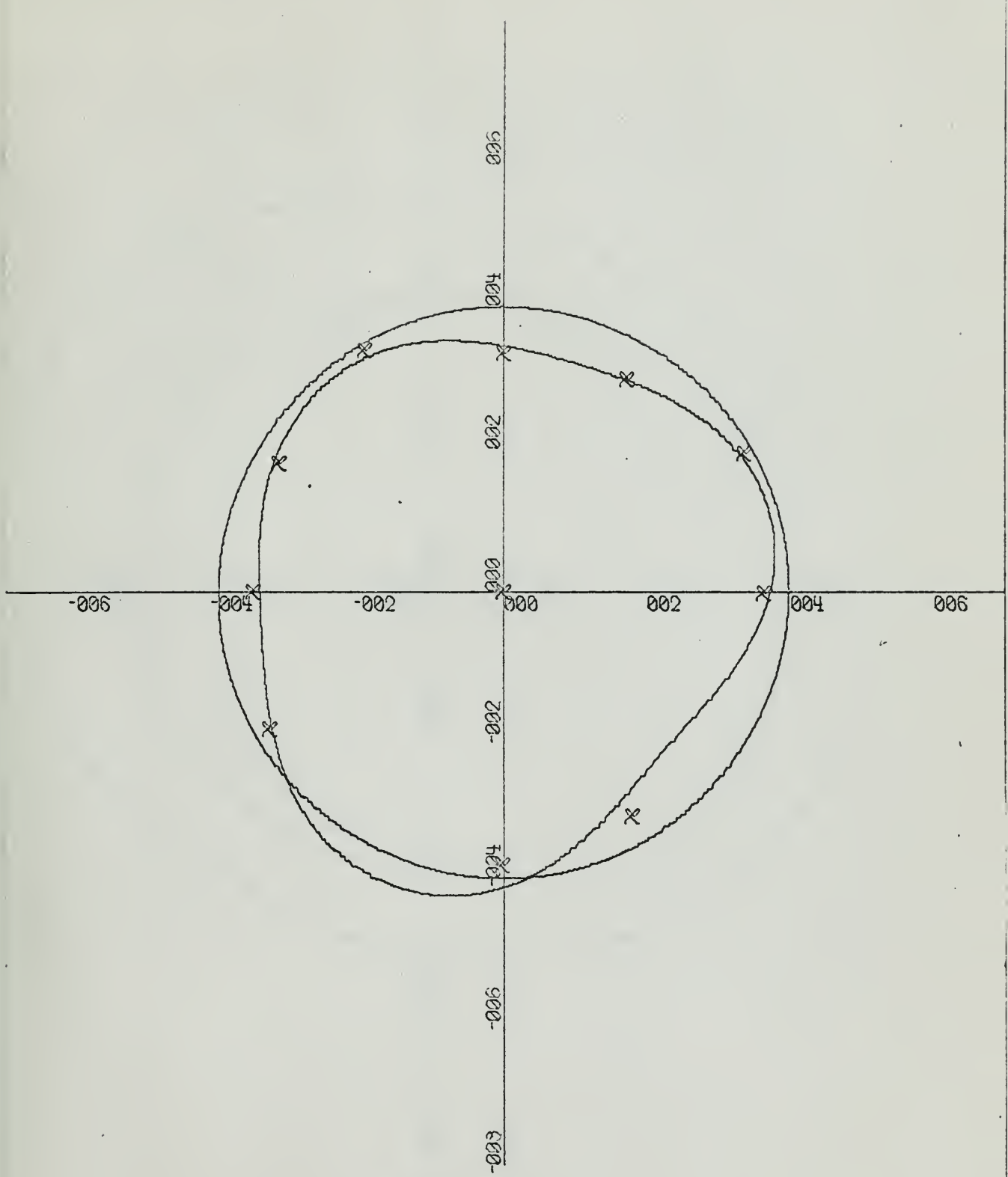


FIGURE E-8

Polar plot of τ_{xy} for rosette ring #4

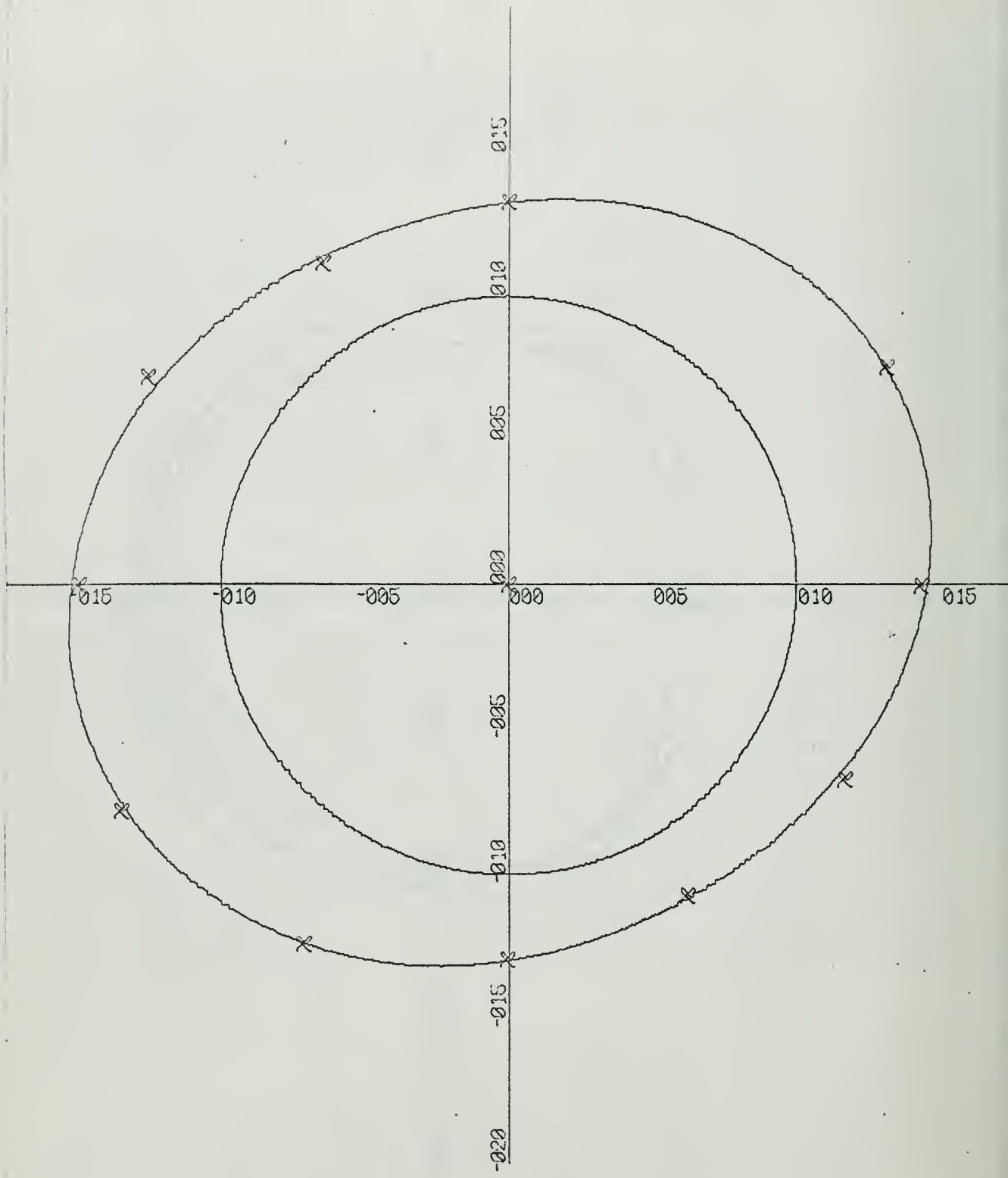


FIGURE E-9

Polar plot of σ_N for rosette ring #5

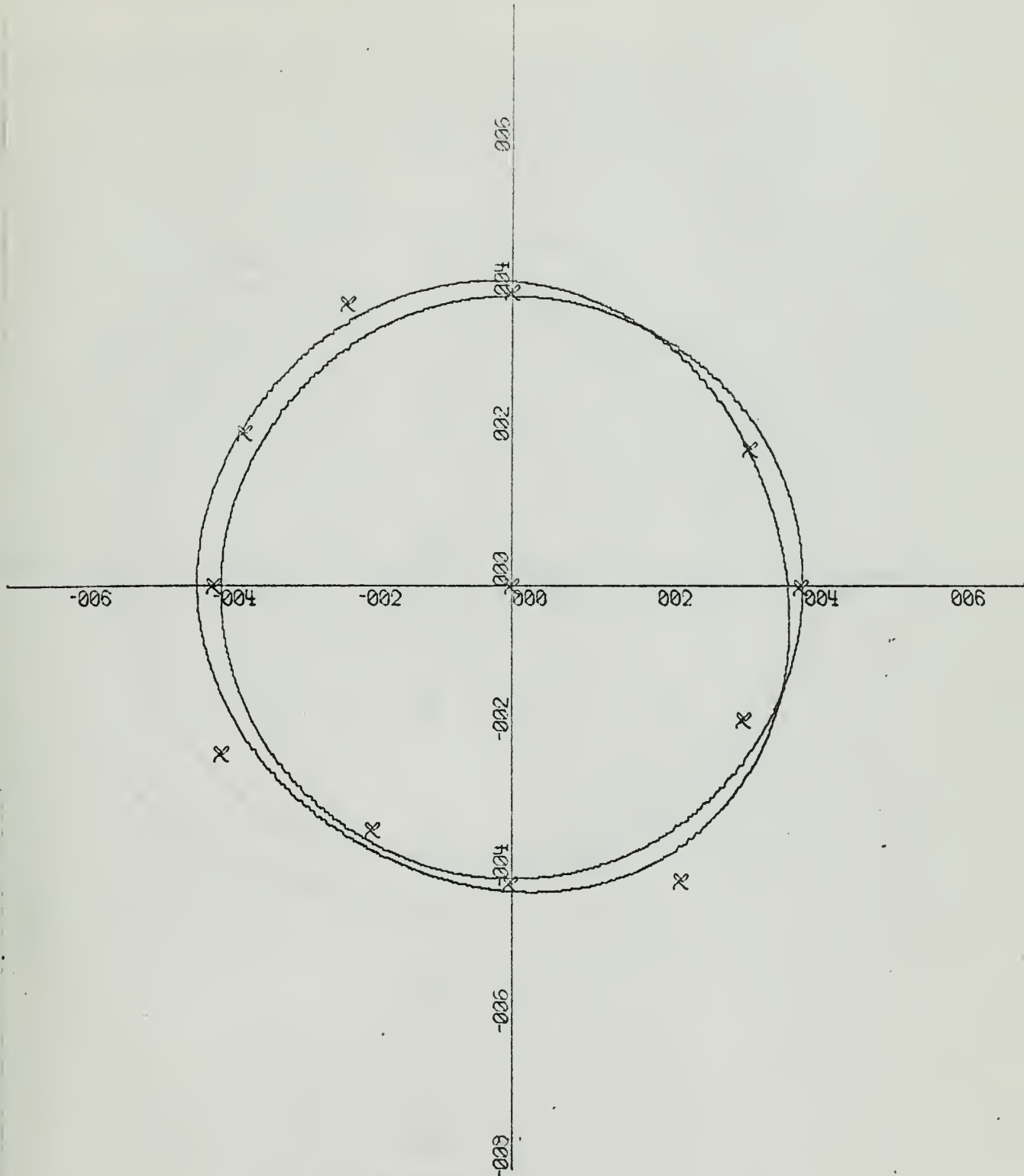


FIGURE E-10

Polar plot of τ_{xy} for rosette ring #5

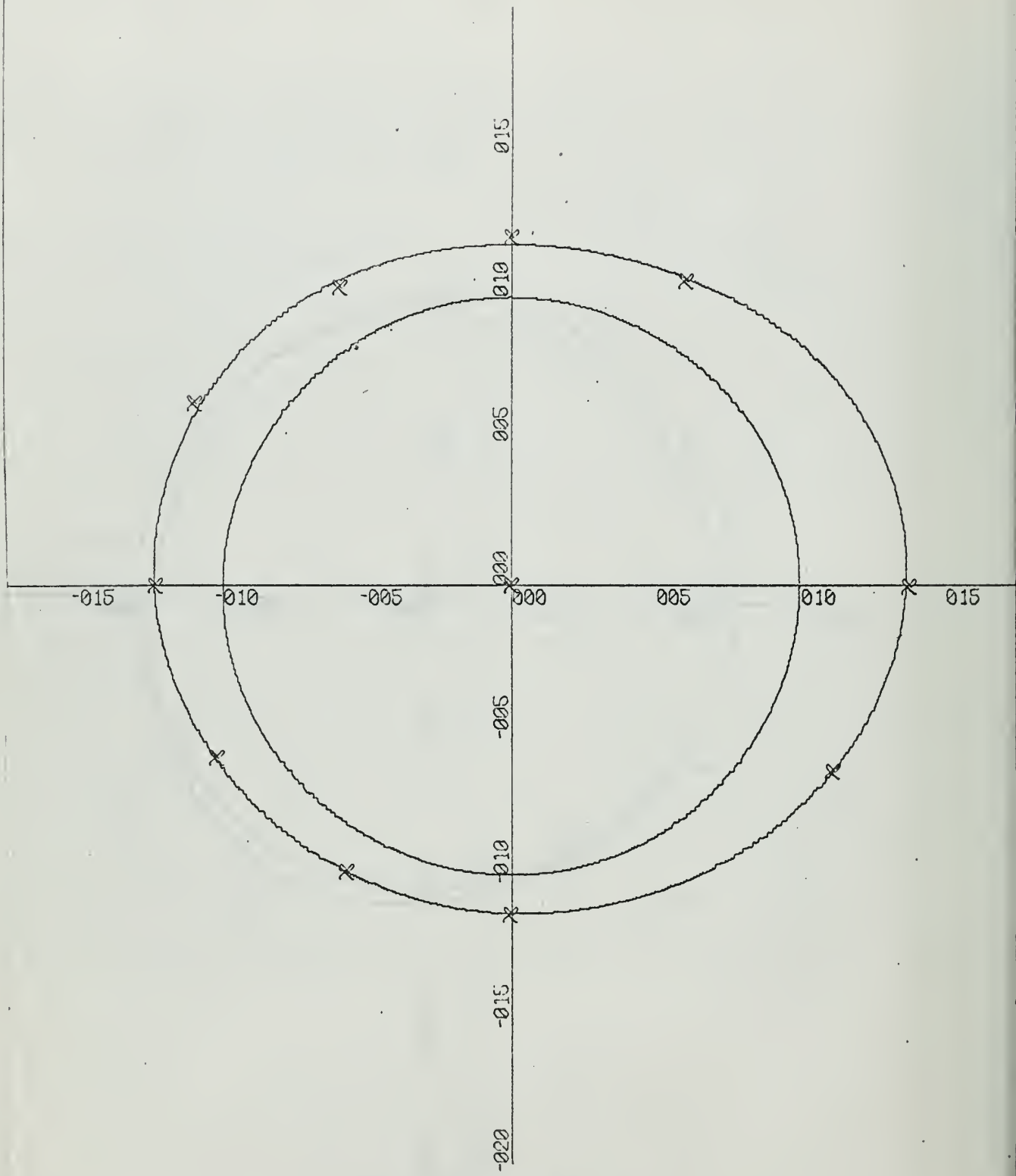


FIGURE E-11

Polar plot of σ_N for rosette ring #6

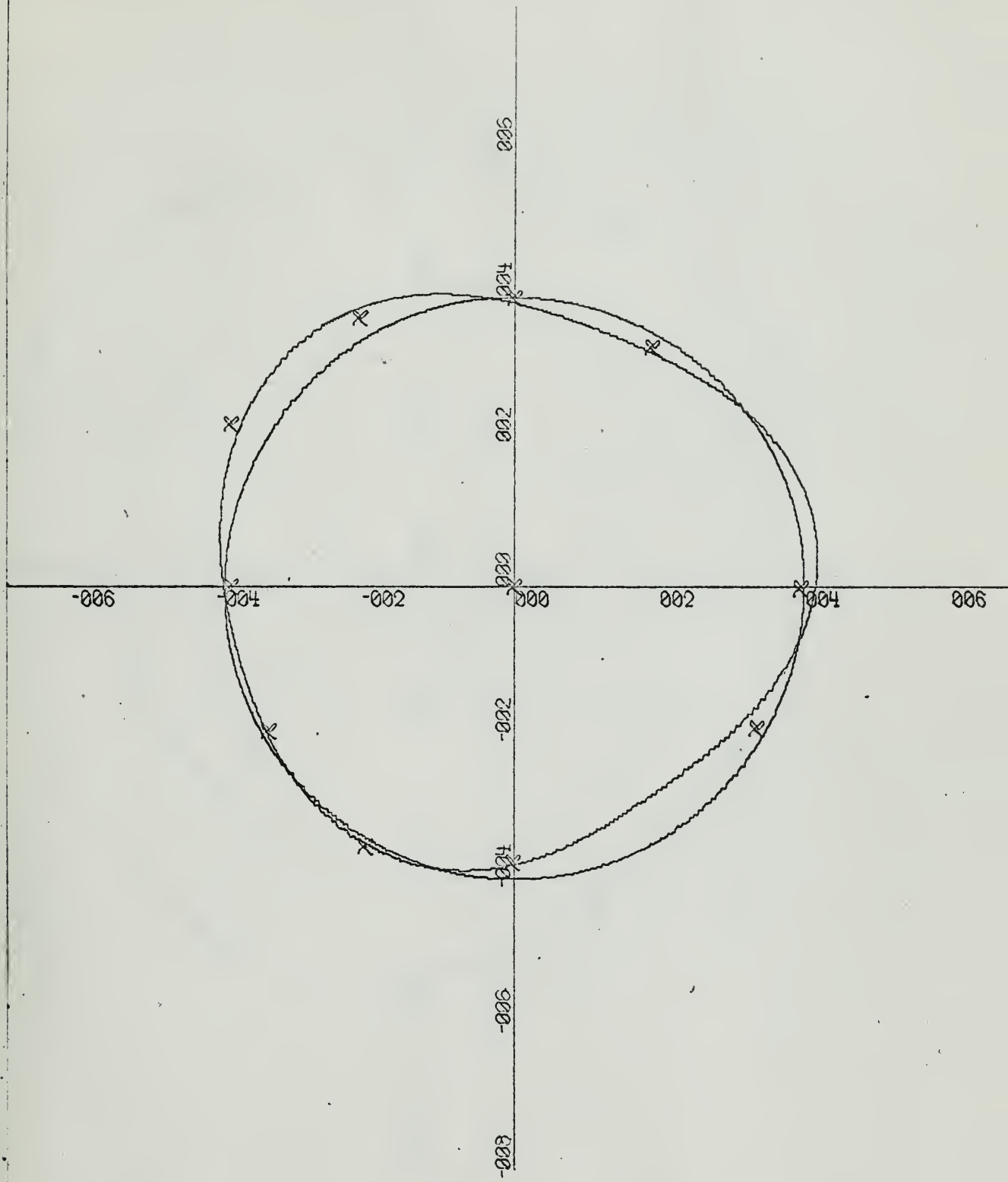


FIGURE E-12

Polar plot of τ_{xy} for rosette ring #6

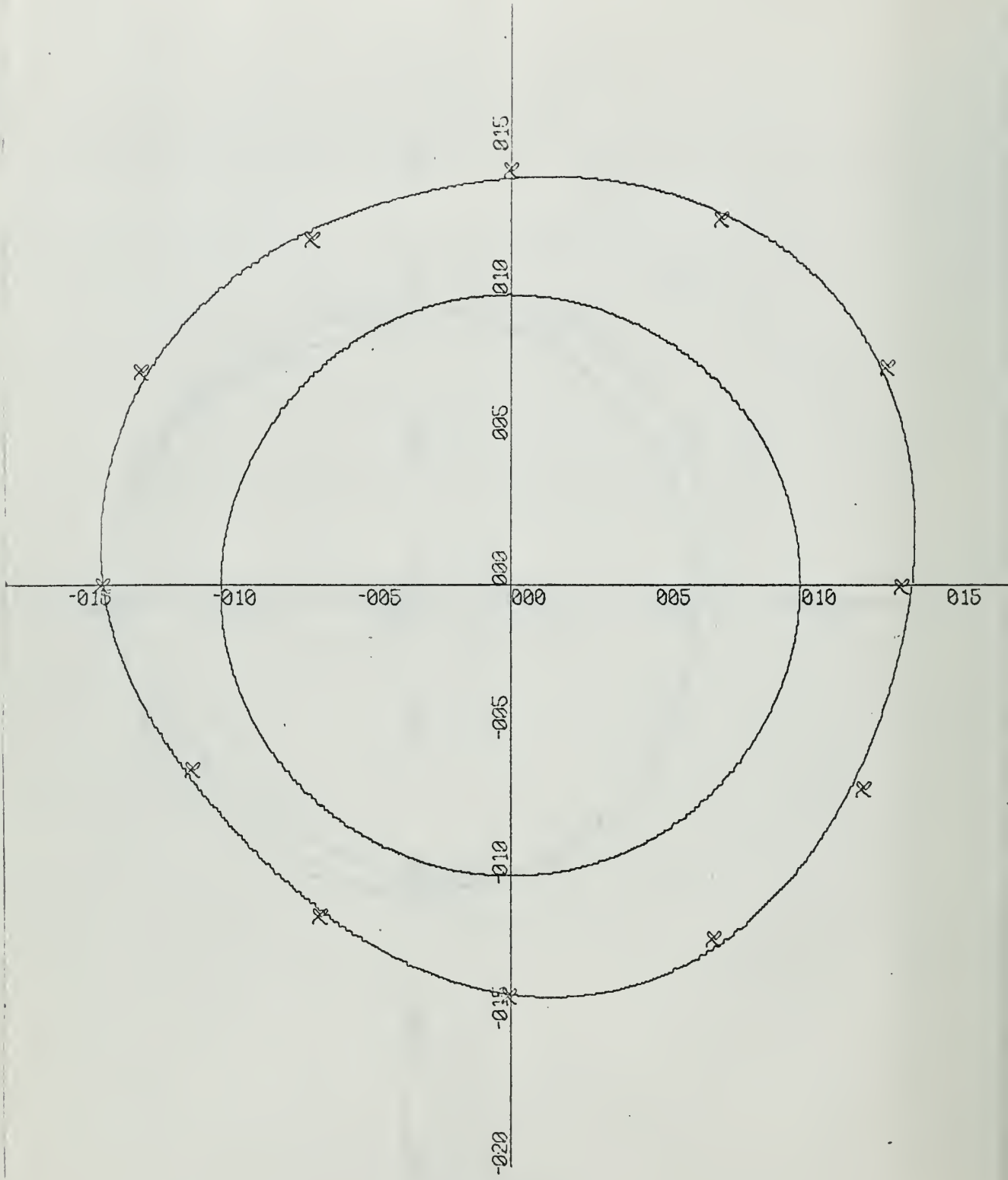


FIGURE E-13

Polar plot of σ_N for rosette ring #7

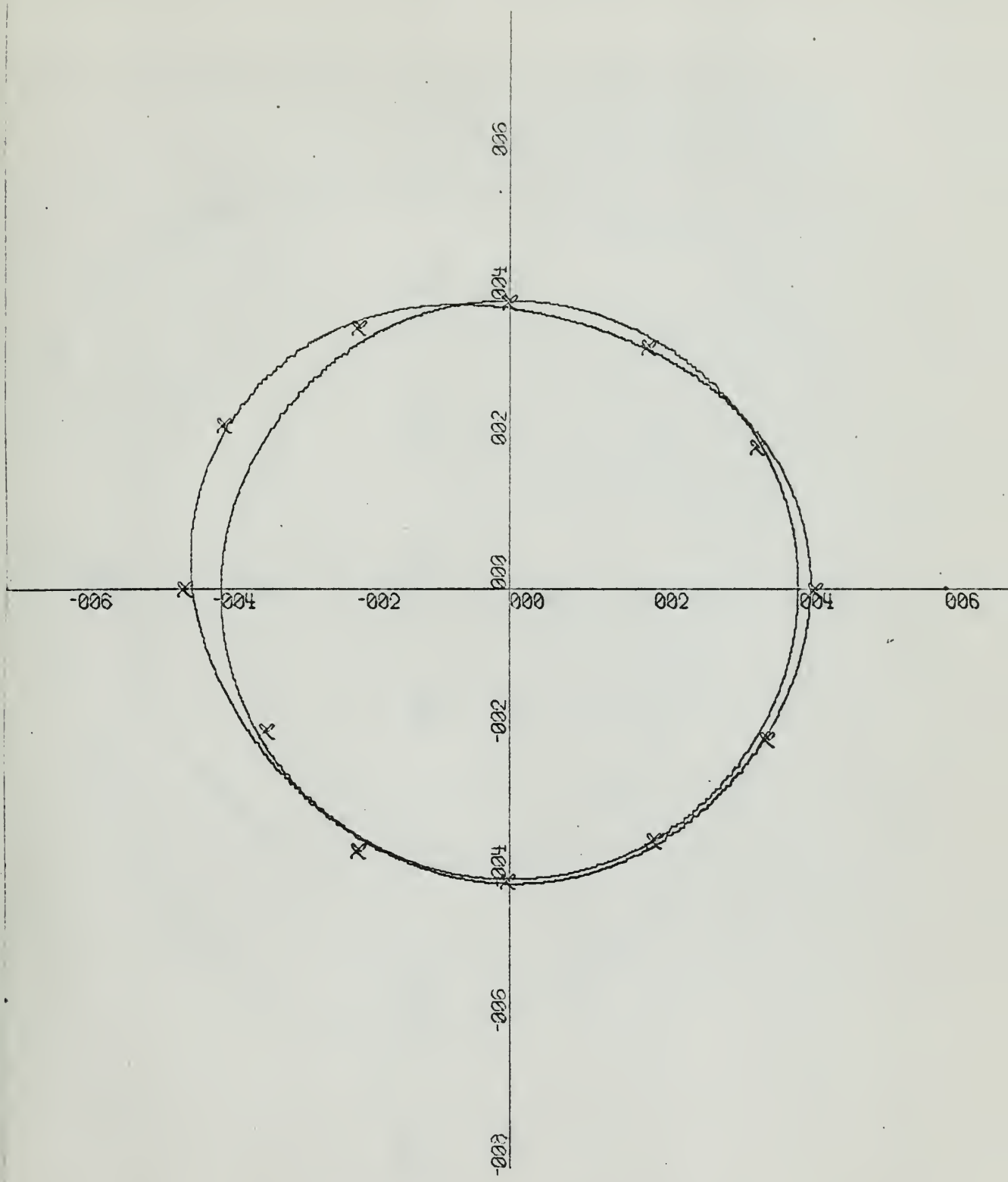


FIGURE E-14

Polar plot of τ_{xy} for rosette ring #7

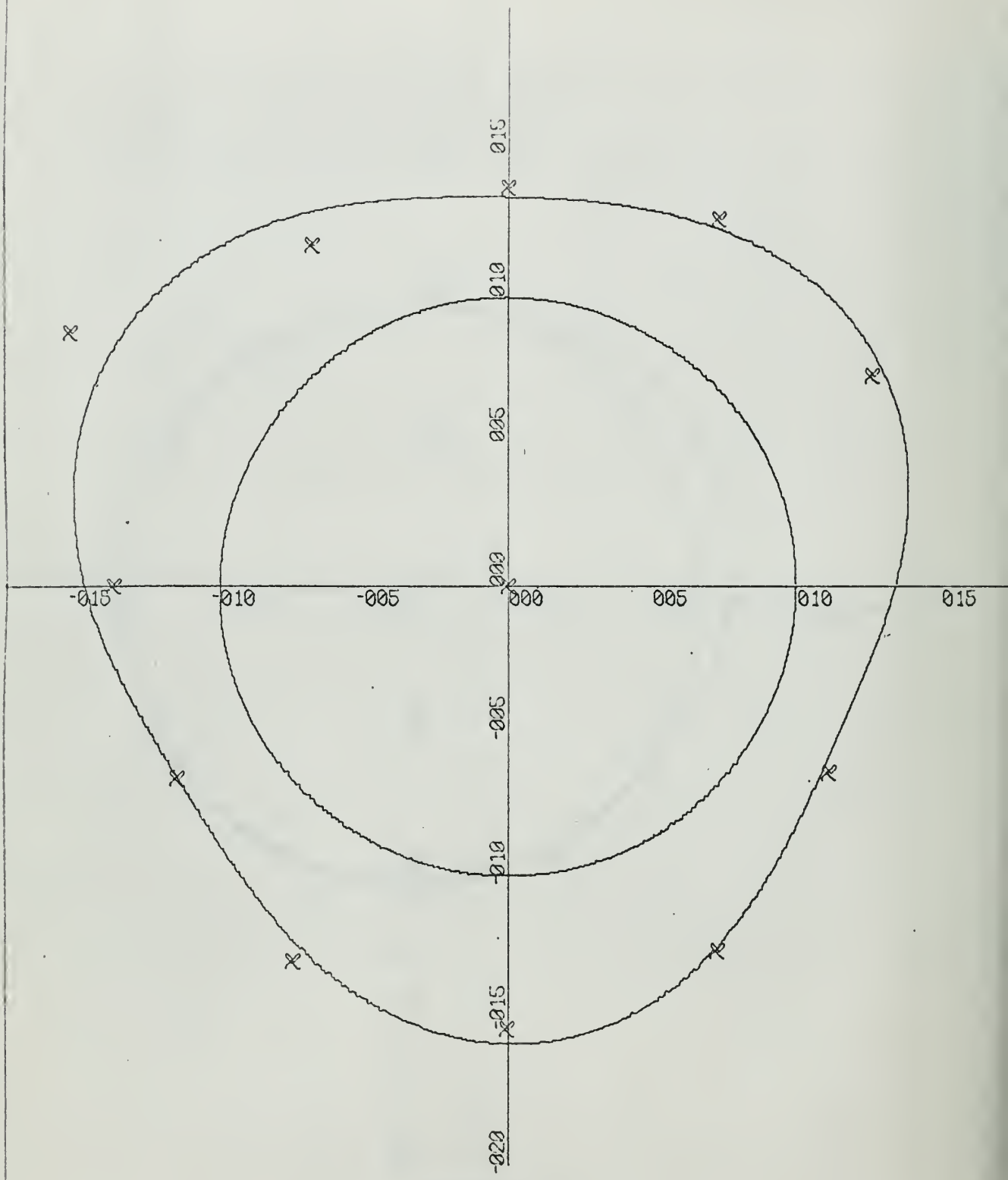


FIGURE E-15

Polar plot of σ_N for rosette ring #8

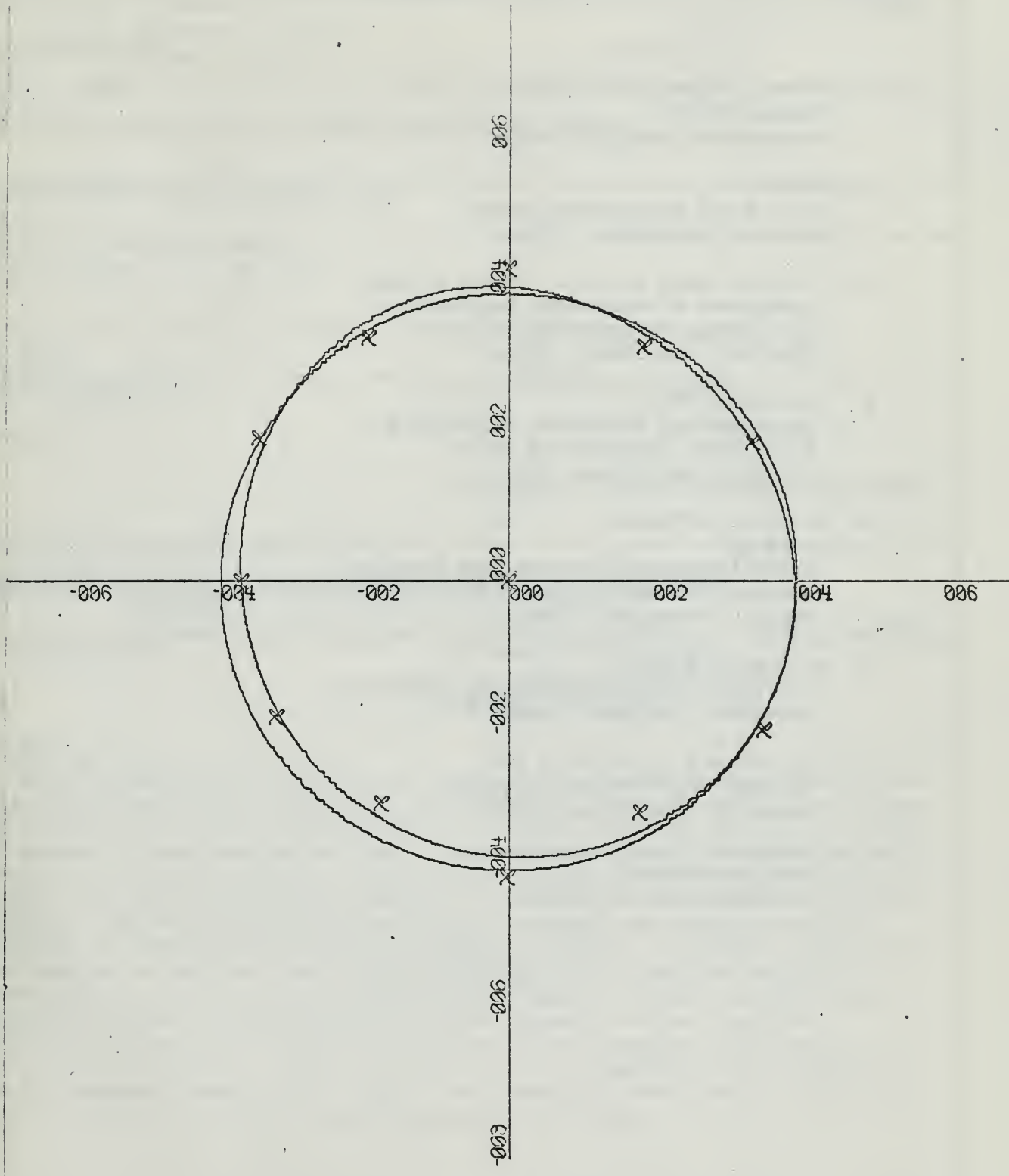


FIGURE E-16

Polar plot of τ_{xy} for rosette ring #8

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2.	Library U. S. Naval Postgraduate School Monterey, California 93940	2
3.	Professor John E. Brock (Thesis Advisor) Department of Mechanical Engineering U. S. Naval Postgraduate School Monterey, California 93940	3
4.	Department Files Department of Mechanical Engineering U. S. Naval Postgraduate School Monterey, California 93940	1
5.	Mr. Carl P. Wilson Code 452 General Dynamics/Electric Boat Division Eastern Point Road Groton, Connecticut 06340	3
6.	Mr. Leslie Kaldor U. S. Navy Marine Engineering Laboratory Annapolis, Maryland 21402	3
7.	Lt. Joseph W. Koch, Jr., USN 451 Dela Vina Avenue, Apt. 415 Monterey, California 93940	2
8.	Commander, Ship Systems Command Navy Department Washington, D. C. 20360	1

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
U. S. NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93940

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP
N/A

3. REPORT TITLE
ANALYSIS OF STRAIN GAGE DATA FROM SUBMARINE MSW PIPING

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
N/A

5. AUTHOR(S) *(Last name, first name, initial)*
KOCH, Joseph W. Jr., LT, USN

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
MAY 1966	102 101	8

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N/A	N/A
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	N/A

10. AVAILABILITY/LIMITATION NOTICES
[redacted] This document has been approved for public release and sale; its distribution is unlimited. #/ Wennikon 12/22/65

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
N/A	U. S. NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA 93940

13. ABSTRACT

This thesis presents a procedural sequence for processing and assessing the integrity of expermental strain gage data taken on piping on which gage rosettes are arranged in circumferential patterns at several sections of a piping system. The method of least squares is used to obtain "best" values for strain per unit loading for each gage element. This data is used to find principle stresses and values of normal and shearing stress on a normal cross section at each rosette location. The data is then smoothed to obtain normal and shearing stress as functions of angular position for each ring of rosette. A computer program has been devised to perform these operations and provide a graphical output to assess degree of fit with experimental data. Finally, manual computation is employed to study the equilibrium of various segments of the piping in evaluating the overall integrity of the data. This procedure and program have been used in evaluating tests performed on the main seawater piping system of the USS Benjamin Franklin (SSBN 640). Some results of the dockside hydrostatic tests are presented herein for purposes of illustration.

DD FORM 1 JAN 64 1473

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
MAIN SEAWATER PIPING						
STRESS ANALYSIS						
STRAIN GAGES						
DATA REDUCTION						
STRAIN GAGE ROSETTE RINGS						
EXPERIMENTS ON PIPING						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.